

全国计算机技术与软件专业技术资格（水平）考试指定用书

软件设计师
2009至2016年试题分析与解答

全国计算机专业技术资格考试办公室 主编

清华大学出版社

全国计算机技术与软件专业技术资格（水平）考试指定用书

软件设计师
2009至2016年试题分析与解答

全国计算机专业技术资格考试办公室 主编

清华大学出版社
北京

内 容 简 介

软件设计师考试是全国计算机技术与软件专业技术资格（水平）考试的中级职称考试，是历年各级考试报名中最大的热点之一。本书汇集了 2009 上半年到 2016 下半年的所有试题和权威的解析，参加考试的考生，认真读懂本书的内容后，将会更加了解考题的思路，对提升自己考试通过率的信心会有极大的帮助。

图书在版编目（CIP）数据

软件设计师 2009 至 2016 年试题分析与解答 / 全国计算机专业技术资格考试办公室主编. —北京：清华大学出版社，2017
（全国计算机技术与软件专业技术资格（水平）考试指定用书）
ISBN 978-7-302-48585-8

Ⅰ. ①软…　Ⅱ. ①全…　Ⅲ. ①软件设计－资格考试－题解　Ⅳ. ①TP311.1-44

中国版本图书馆 CIP 数据核字（2017）第 250008 号

责任编辑：杨如林　柴文强
封面设计：常雪影
责任校对：胡伟民
责任印制：李红英

出版发行：清华大学出版社
网　　址：http://www.tup.com.cn, http://www.wqbook.com
地　　址：北京清华大学学研大厦 A 座　　**邮　　编：**100084
社 总 机：010-62770175　　**邮　　购：**010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质量反馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
印 装 者：清华大学印刷厂
经　　销：全国新华书店
开　　本：185mm×230mm　**印　　张：**52.75　**防伪页：**1　**字　　数：**1160 千字
版　　次：2017 年 12 月第 1 版　　**印　　次：**2017 年 12 月第 1 次印刷
印　　数：1～3000
定　　价：129.00 元

产品编号：075339-01

前　　言

根据国家有关的政策性文件，全国计算机技术与软件专业技术资格（水平）考试（以下简称“计算机软件考试”）已经成为计算机软件、计算机网络、计算机应用、信息系统、信息服务领域高级工程师、工程师、助理工程师、技术员国家职称资格考试。而且，根据信息技术人才年轻化的特点和要求，报考这种资格考试不限学历与资历条件，以不拘一格选拔人才。现在，软件设计师、程序员、网络工程师、数据库系统工程师、系统分析师、系统架构设计师和信息系统项目管理师等资格的考试标准已经实现了中国与日本互认，程序员和软件设计师等资格的考试标准已经实现了中国和韩国互认。

计算机软件考试规模发展很快，年报考规模已经超过 30 万人，二十多年来，累计报考人数超过 470 万人。

计算机软件考试已经成为我国著名的 IT 考试品牌，其证书的含金量之高已得到社会的公认。计算机软件考试的有关信息见网站 www.ruankao.org.cn 中的资格考试栏目。

对考生来说，学习历年试题分析与解答是理解考试大纲的最有效、最具体的途径。

为帮助考生复习备考，全国计算机专业技术资格考试办公室组织编写了软件设计师 2009 至 2016 年的试题分析与解答，以便于考生测试自己的水平，发现自己的弱点，更有针对性、更系统地学习。

计算机软件考试的试题质量高，包括了职业岗位所需的各个方面的知识和技术，不但包括技术知识，还包括法律法规、标准、专业英语、管理等方面的知识；不但注重广度，而且还有一定的深度；不但要求考生具有扎实的基础知识，还要具有丰富的实践经验。

这些试题中，包含了一些富有创意的试题，一些与实践结合得很好的试题，一些富有启发性的试题，具有较高的社会引用率，对学校教师、培训指导者、研究工作者都是很有帮助的。

由于作者水平有限，时间仓促，书中难免有错误和疏漏之处，诚恳地期望各位专家和读者批评指正，对此，我们将深表感激。

编者

2017 年 8 月

目 录

第 1 章　2009 上半年软件设计师上午试题分析与解答

试题（1）

海明校验码是在 n 个数据位之外增设 k 个校验位，从而形成一个 $k+n$ 位的新的码字，使新的码字的码距比较均匀地拉大。n 与 k 的关系是__（1）__。

（1）A. $2^k-1\geqslant n+k$　　B. $2^n-1\leqslant n+k$　　C. $n=k$　　D. $n-1\leqslant k$

试题（1）分析

本题考查校验码方面的基础知识。

海明码是一种多重（复式）奇偶检错编码。它将信息用逻辑形式编码，以便能够检错和纠错。用在海明码中的全部传输码字是由原来的信息和附加的奇偶校验位组成的。每一个这种奇偶位被编在传输码字的特定位置上。推导并使用长度为 n 位的码字的海明码，所需步骤如下：

（1）确定最小的校验位数 k，将它们记成 D_1、D_2、…、D_k，每个校验位符合不同的奇偶测试规定。

（2）原有信息和 k 个校验位一起编成长为 $n+k$ 位的新码字。选择 k 校验位（0 或 1）以满足必要的奇偶条件。

（3）对所接收的信息作所需的 k 个奇偶检查。

（4）如果所有的奇偶检查结果均正确，则认为信息无错误。如果发现有一个或多个错了，则错误的位由这些检查的结果来唯一地确定。

求海明码时的一项基本考虑是确定所需最少的校验位数 k。考虑长度为 n 位的信息，若附加了 k 个校验位，则所发送的总长度为 $n+k$。在接收器中要进行 k 个奇偶检查，每个检查结果或是真或是假。这个奇偶检查的结果可以表示成一个 k 位的二进制，它可以确定最多 2^k 种不同状态。这些状态中必有一个其所有奇偶测试都是真的，它便是判定信息正确的条件。于是剩下的（2^k–1）种状态可以用来判定误码的位置。于是导出以下关系：

$$2^k-1\geqslant n+k$$

参考答案

（1）A

试题（2）

假设某硬盘由 5 个盘片构成(共有 8 个记录面)，盘面有效记录区域的外直径为 30cm，内直径为 10cm，记录位密度为 250 位/mm，磁道密度为 16 道/mm，每磁道分 16 个扇区，每扇区 512 字节，则该硬盘的格式化容量约为__（2）__MB。

（2）A. $\frac{8*(30-10)*10*250*16}{8*1024*1024}$　　B. $\frac{8*(30-10)*10*16*16*512}{2*1024*1024}$

C. $\frac{8*(30-10)*10*250*16*16}{8*1024*1024}$　　D. $\frac{8*(30-10)*16*16*512}{2*1024*1024}$

试题（2）分析

本题考查计算机系统硬件方面磁盘容量的计算。

硬盘容量分为非格式化容量和格式化容量两种，计算公式如下：

非格式化容量＝面数×（磁道数/面）×内圆周长×最大位密度

格式化容量＝面数×（磁道数/面）×（扇区数/道）×（字节数/扇区）

题目中给出硬盘的面数为 8，每面的磁道数为（30–10）×10÷2×16，每磁道扇区数为 16，每扇区 512 字节，因此其格式化容量为

$$\frac{8*(30-10)*10*16*16*512}{2}\text{B}$$

换算成 MB 单位时再除以 1024*1024。

参考答案

（2）B

试题（3）

__（3）__是指按内容访问的存储器。

（3）A. 虚拟存储器　　B. 相联存储器

C. 高速缓存（Cache）　　D. 随机访问存储器

试题（3）分析

本题考查计算机系统存储器方面的基础知识。

计算机系统的存储器按所处的位置可分为内存和外存。按构成存储器的材料可分为磁存储器、半导体存储器和光存储器。按存储器的工作方式可分为读写存储器和只读存储器。按访问方式可分为按地址访问的存储器和按内容访问的存储器。按寻址方式可分为随机存储器、顺序存储器和直接存储器。

相联存储器是一种按内容访问的存储器。

参考答案

（3）B

试题（4）

处理机主要由处理器、存储器和总线组成，总线包括__（4）__。

（4）A. 数据总线、地址总线、控制总线　　B. 并行总线、串行总线、逻辑总线

C. 单工总线、双工总线、外部总线　　D. 逻辑总线、物理总线、内部总线

试题（4）分析

本题考查计算机系统总线和接口方面的基础知识。

广义地讲，任何连接两个以上电子元器件的导线都可以称为总线。通常可分为 4 类：

① 芯片内总线。用于在集成电路芯片内部各部分的连接。

② 元件级总线。用于一块电路板内各元器件的连接。

③ 内总线，又称系统总线。用于构成计算机各组成部分（CPU、内存和接口等）的连接。

④ 外总线，又称通信总线。用计算机与外设或计算机与计算机的连接或通信。

连接处理机的处理器、存储器及其他部件的总线属于内总线，按总线上所传送的内容分为数据总线、地址总线和控制总线。

参考答案

（4）A

试题（5）

计算机中常采用原码、反码、补码和移码表示数据，其中，±0 编码相同的是__（5）__。

（5）A. 原码和补码　　B. 反码和补码

C. 补码和移码　　D. 原码和移码

试题（5）分析

本题考查计算机系统数据编码基础知识。

设机器字长为 n（即采用 n 个二进制位表示数据），最高位是符号位，0 表示正号，1 表示负号。

原码表示方式下，除符号位外，n–1 位表示数值的绝对值。因此，n 为 8 时，$[+0]_{原}$ = 0 0000000，$[-0]_{原}$=1 0000000。

正数的反码与原码相同，负数的反码则是其绝对值按位求反。n 为 8 时，数值 0 的反码表示有两种形式：$[+0]_{反}$=0 0000000，$[-0]_{反}$=1 1111111。

正数的补码与其原码和反码相同，负数的补码则等于其反码的末尾加 1。在补码表示中，0 有唯一的编码：$[+0]_{补}$=0 0000000，$[-0]_{补}$=00000000。

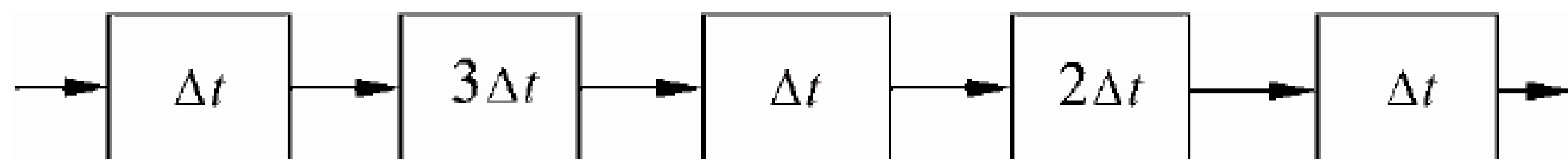
移码表示法是在数 X 上增加一个偏移量来定义的，常用于表示浮点数中的阶码。机器字长为 n 时，在偏移量为 2^{n-1} 的情况下，只要将补码的符号位取反便可获得相应的移码表示。

参考答案

（5）C

试题（6）

某指令流水线由 5 段组成，第 1、3、5 段所需时间为Δt，第 2、4 段所需时间分别为 $3\Delta t$、$2\Delta t$，如下图所示，那么连续输入 n 条指令时的吞吐率（单位时间内执行的指令个数）TP 为__（6）__。



（6）A. $\frac{n}{5*(3+2)\Delta t}$ B. $\frac{n}{(3+3+2)\Delta t+3(n-1)\Delta t}$

C. $\frac{n}{(3+2)\Delta t+(n-3)\Delta t}$ D. $\frac{n}{(3+2)\Delta t+5*3\Delta t}$

试题（6）分析

本题考查计算机系统流水线方面的基础知识。

吞吐率和建立时间是使用流水线技术的两个重要指标。吞吐率是指单位时间里流水线处理机流出的结果数。对指令而言，就是单位时间里执行的指令数。流水线开始工作时，需经过一定时间才能达到最大吞吐率，这就是建立时间。若 m 个子过程所用时间一样，均为 Δt_0，则建立时间 $T_0=m\Delta t_0$。

本题目中，连续输入 n 条指令时，第 1 条指令需要的时间为 $(1+3+1+2+1)\Delta t$，之后，每隔 $3\Delta t$ 便完成 1 条指令，即流水线一旦建立好，其吞吐率为最长子过程所需时间的倒数。综合 n 条指令的时间为 $(1+3+1+2+1)\Delta t+(n-1)\times 3\Delta t$，因此吞吐率为

$$\frac{n}{(3+3+2)\Delta t+3(n-1)\Delta t}$$

参考答案

（6）B

试题（7）

下面关于漏洞扫描系统的叙述，错误的是<u>（7）</u>。

（7）A. 漏洞扫描系统是一种自动检测目标主机安全弱点的程序

B. 黑客利用漏洞扫描系统可以发现目标主机的安全漏洞

C. 漏洞扫描系统可以用于发现网络入侵者

D. 漏洞扫描系统的实现依赖于系统漏洞库的完善

试题（7）分析

本题考查漏洞扫描系统的基本概念。

漏洞扫描系统是一种自动检测目标主机安全弱点的程序，漏洞扫描系统的原理是根据系统漏洞库对系统可能存在的漏洞进行一一验证。黑客利用漏洞扫描系统可以发现目标主机的安全漏洞从而有针对性的对系统发起攻击；系统管理员利用漏洞扫描系统可以查找系统中存在的漏洞并进行修补从而提高系统的可靠性。漏洞扫描系统不能用于发现网络入侵者，用于检测网络入侵者的系统称为入侵检测系统。

参考答案

（7）C

试题（8）

网络安全包含了网络信息的可用性、保密性、完整性和网络通信对象的真实性。其中，数字签名是对<u>（8）</u>的保护。

（8）A. 可用性　　B. 保密性　　C. 连通性　　D. 真实性

试题（8）分析

本题考查网络安全方面的基础知识。

数字签名（Digital Signature）技术是不对称加密算法的典型应用。数字签名的应用过程是：数据源发送方使用自己的私钥对数据校验和或其他与数据内容有关的变量进行加密处理，完成对数据的合法“签名”；数据接收方则利用对方的公钥来解读收到的“数字签名”，并将解读结果用于对数据完整性的检验，以确认签名的合法性。数字签名技术是在网络系统虚拟环境中确认身份的重要技术，完全可以代替现实过程中的“亲笔签字”，在技术和法律上有保证，可见数字签名是对签名真实性的保护。

参考答案

（8）D

试题（9）

计算机感染特洛伊木马后的典型现象是__（9）__。

（9）A. 程序异常退出　　B. 有未知程序试图建立网络连接
C. 邮箱被垃圾邮件填满　　D. Windows 系统黑屏

试题（9）分析

本题考查计算机病毒相关知识。

特洛伊木马是一种通过网络传播的病毒，分为客户端和服务器端两部分，服务器端位于被感染的计算机，特洛伊木马服务器端运行后会试图建立网络连接，所以计算机感染特洛伊木马后的典型现象是有未知程序试图建立网络连接。

参考答案

（9）B

试题（10）

关于软件著作权产生的时间，下面表述正确的是__（10）__。

（10）A. 自作品首次公开发表时
B. 自作者有创作意图时
C. 自作品得到国家著作权行政管理部门认可时
D. 自作品完成创作之日

试题（10）分析

本题考查知识产权中关于软件著作权方面的知识。

在我国，软件著作权采用“自动保护”原则。《计算机软件保护条例》第十四条规定：“软件著作权自软件开发完成之日起产生。”即软件著作权自软件开发完成之日起自动产生，不论整体还是局部，只要具备了软件的属性即产生软件著作权，既不要求履行任何形式的登记或注册手续，也无须在复制件上加注著作权标记，也不论其是否已经发表都依法享有软件著作权。

一般来讲，一个软件只有开发完成并固定下来才能享有软件著作权。如果一个软件一直处于开发状态中，其最终的形态并没有固定下来，则法律无法对其进行保护。因此，条例（法律）明确规定软件著作权自软件开发完成之日起产生。当然，现在的软件开发经常是一项系统工程，一个软件可能会有很多模块，而每一个模块能够独立完成某一项功能。自该模块开发完成后就产生了著作权。所以说，自该软件开发完成后就产生了著作权。

参考答案

（10）D

试题（11）

程序员甲与同事乙在乙家探讨甲近期编写的程序，甲表示对该程序极不满意，说要弃之重写，并将程序手稿扔到乙家垃圾筒。后来乙将甲这一程序稍加修改，并署乙名发表。以下说法正确的是__（11）__。

（11）A. 乙的行为侵犯了甲的软件著作权

B. 乙的行为没有侵犯甲的软件著作权，因为甲已将程序手稿丢弃

C. 乙的行为没有侵犯甲的著作权，因为乙已将程序修改

D. 甲没有发表该程序并弃之，而乙将程序修改后发表，故乙应享有著作权

试题（11）分析

本题考查知识产权中关于软件著作权方面的知识。

著作权因作品的完成而自动产生，不必履行任何形式的登记或注册手续，也不论其是否已经发表，所以甲对该软件作品享有著作权。乙未经甲的许可擅自使用甲的软件作品的行为，侵犯了甲的软件著作权。

参考答案

（11）A

试题（12）

PC 处理的音频信号主要是人耳能听得到的音频信号，它的频率范围是__（12）__。

（12）A. 300Hz～3400Hz　　B. 20Hz～20kHz

C. 10Hz～20kHz　　D. 20Hz～44kHz

试题（12）分析

本题考查多媒体中关于音频信号方面的基础知识。

声音信号由许多频率不同的信号组成，通常称为复合信号，而把单一频率的信号称为分量信号。声音信号的一个重要参数就是带宽（Bandwidth），它用来描述组成声音的信号的频率范围。

声音信号的频率是指声波每秒钟变化的次数，用 Hz 表示。人们把频率小于20Hz 的声波信号称为亚音信号（也称次音信号）；频率范围为 20Hz～20kHz 的声波信号称为音频信号；高于 20kHz 的信号称为超音频信号（也称超声波）。

PC 处理的音频信号主要是人耳能听得到的音频信号（audio），它的频率范围是 20～20kHz。可听声包括：

- 话音（也称语音）：人的说话声，频率范围通常为 300～3400Hz。
- 音乐：由乐器演奏形成（规范的符号化声音），其带宽可达到 20～20kHz。
- 其他声音：如风声、雨声、鸟叫声和汽车鸣笛声等，它们起着效果声或噪声的作用，其带宽范围也是 20～20kHz。

参考答案

（12）B

试题（13）

多媒体计算机图像文件格式分为静态图像文件格式和动态图像文件格式，（13）属于静态图像文件格式。

（13）A. MPG　　B. AVS　　C. JPG　　D. AVI

试题（13）分析

本题考查多媒体中关于文件格式方面的基础知识。

计算机中使用的图像文件格式大体上可分为图像文件格式和动态图像文件格式两大类，每类又有很多种。JPEG 是由 ISO 和 IEC 两个组织机构联合组成的一个专家组，负责制定静态和数字图像数据压缩编码标准，这个专家组地区性的算法称为 JPEG 算法，并且成为国际上通用的标准，因此又称为 JPEG 标准。JPEG 是一个适用范围很广的静态图像数据压缩标准，既可用于灰度图像又可用于彩色图像。MPEG 文件格式是运动图像压缩算法的国际标准，它包括 MPEG 视频、MPEG 音频和 MPEG 系统（视频、音频同步）三个部分。MPEG 压缩标准是针对运动图像设计的，其基本方法是：单位时间内采集并保存第一帧信息，然后只存储其余帧对第一帧发生变化的部分，从而达到压缩的目的。MPEG 的平均压缩比为 50:1，最高可达 200:1，压缩效率非常高，同时图像和音响的质量也非常好，并且在 PC 上有统一的标准格式，兼容性相当好。AVI 是 Microsoft 公司开发的一种符合 RIFF 文件规范的数字音频与视频文件格式，Windows、OS/2 等多数操作系统直接支持。AVI 格式允许视频和音频交错在一起同步播放，支持 256 色和 RLE 压缩，但 AVI 文件并未限定压缩标准。AVI 文件目前主要应用在多媒体光盘上，用来保存电影、电视等各种影像信息，有时也出现在因特网上，供用户下载、欣赏新影片的片段。

参考答案

（13）C

试题（14）

计算机获取模拟视频信息的过程中首先要进行（14）。

（14）A. A/D 变换　　B. 数据压缩　　C. D/A 变换　　D. 数据存储

试题（14）分析

本题考查多媒体中关于模拟视频信息处理方面的基础知识。

模拟视频信号进入计算机时，首先需要解决模拟视频信息的数字化问题。与音频数字化一样，视频数字化的目的是将模拟信号经 A/D 转换和彩色空间变换等过程，转换成计算机可以显示和处理的数字信号。由于电视和计算机的显示机制不同，因此要在计算机上显示视频图像需要作许多处理。例如，电视是隔行扫描，计算机的显示器通常是逐行扫描；电视是亮度（Y）和色度（C）的复合编码，而 PC 的显示器工作在 RGB 空间；电视图像的分辨率和显示屏的分辨率也各不相同等。这些问题在电视图像数字化过程中都需考虑。一般，对模拟视频信息进行数字化采取如下方式：

（1）先从复合彩色电视图像中分离出彩色分量，然后数字化。目前市场上的大多数电视信号都是复合的全电视信号，如录像带、激光视盘等存储设备上的电视信号。对这类信号的数字化，通常是将其分离成 YUV、YIQ 或 RGB 彩色空间的分量信号，然后用 3 个 A/D 转换器分别进行数字化。这种方式称为复合数字化。

（2）先对全彩色电视信号数字化，然后在数字域中进行分离，以获得 YUV、YIQ 或 RGB 分量信号。用这种方法对电视图像数字化时，只需一个高速 A/D 转换器。这种方式称为分量数字化。

视频信息数字化的过程比声音复杂一些，它是以一幅幅彩色画面为单位进行的。分量数字化方式是使用较多的一种方式。电视信号使用的彩色空间是 YUV 空间，即每幅彩色画面有亮度（Y）和色度（U、V）3 个分量，对这 3 个分量需分别进行取样和量化，得到一幅数字图像。由于人眼对色度信号的敏感程度远不如对亮度信号那么灵敏，因此色度信号的取样频率可以比亮度信号的取样频率低一些，以减少数字视频的数据量。

数字图像数据的数据量大，而数字视频信息的数据量就更加突出。例如，每帧 352×240 像素点，图像深度 16 位的图像，其数据量约为 1.3Mb，每秒 30 帧，其数据量就高达 40Mb/s，这样大的数据量无论是传输、存储还是处理，都是极大的负担。

参考答案

（14）A

试题（15）

在采用面向对象技术构建软件系统时，很多敏捷方法都建议的一种重要的设计活动是__（15）__，它是一种重新组织的技术，可以简化构件的设计而无须改变其功能或行为。

（15）A. 精化　　B. 设计类　　C. 重构　　D. 抽象

试题（15）分析

本题考查采用敏捷方法进行软件开发。敏捷方法中，重构是一种重新组织技术，重新审视需求和设计，重新明确地描述它们以符合新的和现有的需求，可以简化构件的设计而无须改变其功能或行为。

参考答案

（15）C

试题（16）

一个软件开发过程描述了“谁做”“做什么”“怎么做”和“什么时候做”，RUP 用 (16) 来表述“谁做”。

（16）A. 角色　　B. 活动　　C. 制品　　D. 工作流

试题（16）分析

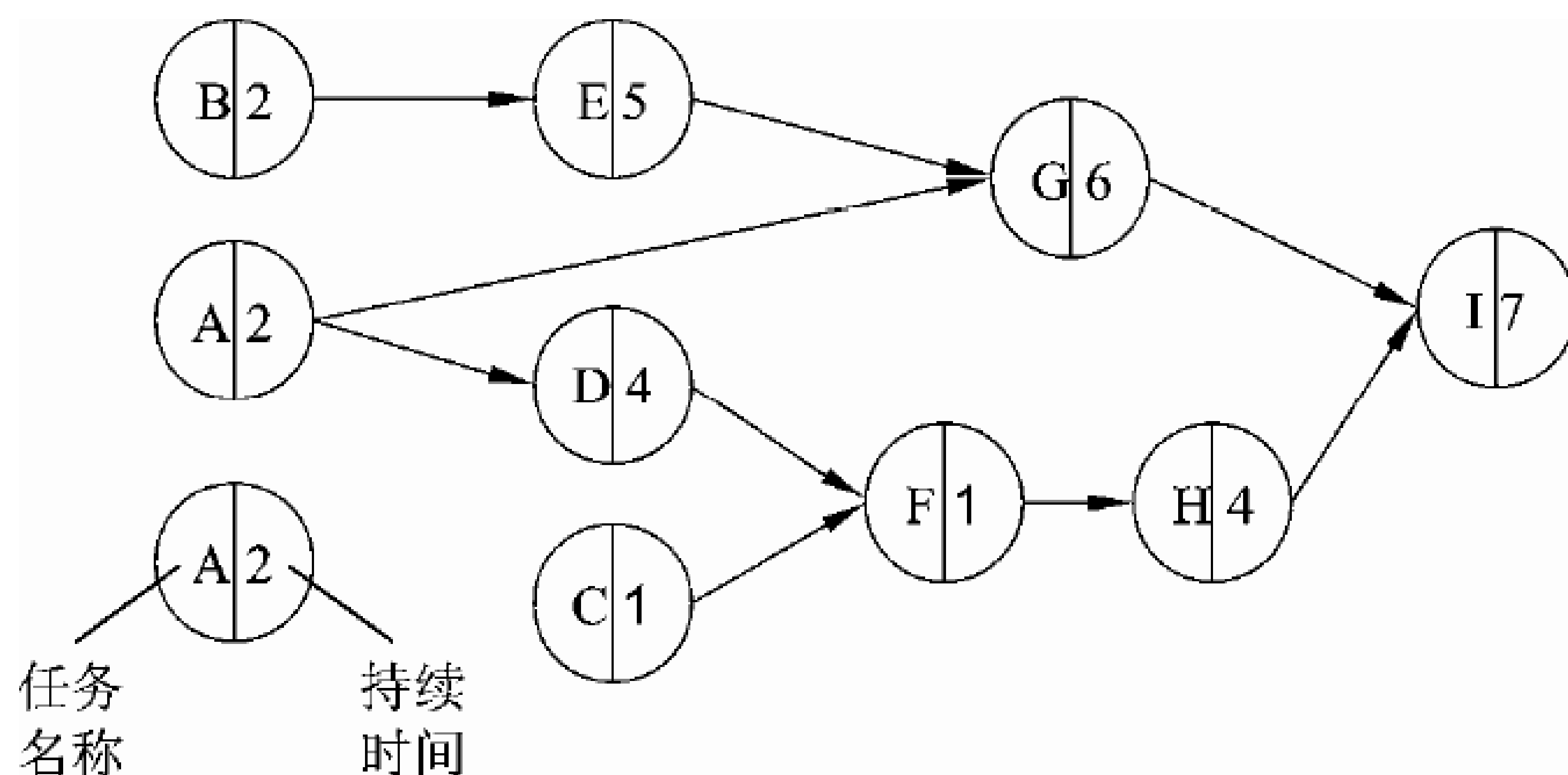
本题考查 RUP 对软件开发过程的描述。RUP 应用了角色、活动、制品和工作流 4 种重要的模型元素，其中角色表述“谁做”，制品表述“做什么”，活动表述“怎么做”，工作流表述“什么时候做”。

参考答案

（16）A

试题（17）、（18）

某项目主要由 A～I 任务构成，其计划图（如下图所示）展示了各任务之间的前后关系以及每个任务的工期（单位：天），该项目的关键路径是 (17) 。在不延误项目总工期的情况下，任务 A 最多可以推迟开始的时间是 (18) 天。



（17）A. A→G→I　　B. A→D→F→H→I
　　C. B→E→G→I　　D. C→F→H→I

（18）A. 0　　B. 2　　C. 5　　D. 7

试题（17）、（18）分析

本题考查项目计划的关键路径和松弛时间。图中任务流 A→G→I 的持续时间为 15；任务流 A→D→F→H→I 的持续时间为 18；任务流 B→E→G→I 的持续时间为 20；任务流 C→F→H→I 的持续时间为 13。因此关键路径为 B→E→G→I，其持续时间是 20。任务 A 处于任务流 A→G→I 和任务流 A→D→F→H→I 中，分别持续时间为 15 和 18，因此任务 A 的可延迟开始时间为 2。

参考答案

（17）C　（18）B

试题（19）

软件风险一般包含＿（19）＿两个特性。

（19）A. 救火和危机管理　B. 已知风险和未知风险
C. 不确定性和损失　D. 员工和预算

试题（19）分析

本题考查软件风险的特性。软件风险一般包括不确定性和损失两个特性，其中不确定性是指风险可能发生，也可能不发生；损失是当风险确实发生时，会引起的不希望的后果和损失。救火和危机管理是对不适合但经常采用的软件风险管理策略。已知风险和未知风险是对软件风险进行分类的一种方式。员工和预算是在识别项目风险时需要识别的因素。

参考答案

（19）C

试题（20）

函数调用时，基本的参数传递方式有传值与传地址两种，＿（20）＿。

（20）A. 在传值方式下，形参将值传给实参
B. 在传值方式下，实参不能是数组元素
C. 在传地址方式下，形参和实参间可以实现数据的双向传递
D. 在传地址方式下，实参可以是任意的变量和表达式

试题（20）分析

本题考查程序语言基础知识。

函数调用时基本的参数传递方式有传值与传地址两种，在传值方式下是将实参的值传递给形参，因此实参可以是表达式（或常量），也可以是变量（或数组元素），这种信息传递是单方向的，形参不能再将值传回给实参。在传地址方式下，需要将实参的地址传递给形参，因此，实参必须是变量（数组名或数组元素），不能是表达式（或常量）。这种方式下，被调用函数中对形式参数的修改实际上就是对实际参数的修改，因此客观上可以实现数据的双向传递。

参考答案

（20）C

试题（21）

已知某高级语言源程序 A 经编译后得到机器 C 上的目标程序 B，则＿（21）＿。

（21）A. 对 B 进行反编译，不能还原出源程序 A
B. 对 B 进行反汇编，不能得到与源程序 A 等价的汇编程序代码
C. 对 B 进行反编译，得到的是源程序 A 的变量声明和算法流程

D. 对 A 和 B 进行交叉编译，可以产生在机器 C 上运行的动态链接库

试题（21）分析

本题考查程序语言方面的基础知识。

编译是将高级语言源程序翻译成机器语言程序（汇编形式或机器代码形式），反编译是编译的逆过程。反编译通常不能把可执行文件还原成高级语言源代码，只能转换成功能上等价的汇编程序。

参考答案

（21）A

试题（22）

下面关于程序语言的叙述，错误的是__（22）__。

（22）A. 脚本语言属于动态语言，其程序结构可以在运行中改变

B. 脚本语言一般通过脚本引擎解释执行，不产生独立保存的目标程序

C. php、JavaScript 属于静态语言，其所有成分可在编译时确定

D. C 语言属于静态语言，其所有成分可在编译时确定

试题（22）分析

本题考查程序语言基础知识。

动态语言是指程序在运行时可以改变其结构，例如新的函数可以被引进、已有的函数可以被删除等在结构上的变化等。动态语言的类型检查是在运行时进行的，其优点是方便阅读，不需要写非常多的与类型相关的代码；缺点是不方便调试，命名不规范时会读不懂、不利于理解等。

脚本语言代表一套与系统程序设计语言不同的协定。它们牺牲执行速度和与系统程序设计语言相关的类型长度而提供更高的编程创作能力和软件重用。脚本语言更适合在联系复杂的应用程序中进行胶着（粘合）。为了简化连接组件的工作，脚本语言被设计为无类型的，脚本语言一般是面向字符的，因为字符为许多不同的事物提供了一致的描述。

事实上，脚本语言都是动态语言，而动态语言都是解释型语言，不管它们是否是面向对象的语言。

参考答案

（22）C

试题（23）、（24）

在 Windows XP 操作系统中，用户利用“磁盘管理”程序可以对磁盘进行初始化、创建卷，__（23）__。通常将“C:\Windows\myprogram.exe”文件设置成只读和隐藏属性，以便控制用户对该文件的访问，这一级安全管理称之为__（24）__安全管理。

（23）A. 但只能使用 FAT 文件系统格式化卷

B. 但只能使用 FAT 32 文件系统格式化卷

C. 但只能使用 NTFS 文件系统格式化卷

D. 可以选择使用 FAT、FAT32 或 NTFS 文件系统格式化卷

（24）A. 文件级　　B. 目录级　　C. 用户级　　D. 系统级

试题（23）、（24）分析

本题考查对 Windows XP 操作系统应用的掌握程度。

试题（23）的正确答案是 D，因为 Windows XP 操作系统支持FAT、FAT32 或 NTFS 文件系统，所以利用“磁盘管理”程序可以对磁盘进行初始化、创建卷，并可以选择使用 FAT、FAT32 或 NTFS 文件系统格式化卷。

试题（24）的正确答案是 A。分析如下：文件级安全管理是通过系统管理员或文件主对文件属性的设置来控制用户对文件的访问。通常可设置以下几种属性：

- 只执行：只允许用户执行该文件，主要针对.exe 和.com 文件。
- 隐含：指示该文件为隐含属性文件。
- 索引：指示该文件是索引文件。
- 修改：指示该文件自上次备份后是否还被修改。
- 只读：只允许用户读该文件。
- 读/写：允许用户对文件进行读和写。
- 共享：指示该文件是可读共享的文件。
- 系统：指示该文件是系统文件。

用户对文件的访问将由用户访问权、目录访问权限及文件属性三者的权限所确定。或者说是有效权限和文件属性的交集。例如对于只读文件，尽管用户的有效权限是读/写，但都不能对只读文件进行修改、更名和删除。对于一个非共享文件，将禁止在同一时间内由多个用户对它们进行访问。通过上述四级文件保护措施，可有效地保护文件。因此将“C:\Windows\myprogram.exe”文件设置成只读和隐藏属性，以便控制用户对该文件的访问，这一级安全管理称之为文件级安全管理。

参考答案

（23）D　（24）A

试题（25）

在移臂调度算法中，__（25）__算法可能会随时改变移动臂的运动方向。

（25）A. 电梯调度和先来先服务

B. 先来先服务和最短寻找时间优先

C. 单向扫描和先来先服务

D. 电梯调度和最短寻找时间优先

试题（25）分析

本题考查对磁盘调度方面基本知识掌握的程度。

因为先来先服务是谁先请求先满足谁的请求，而最短寻找时间优先是根据当前磁臂到要请求访问磁道的距离，谁短满足谁的请求，故先来先服务和最短寻找时间优先算法

可能会随时改变移动臂的运动方向。

参考答案

（25）B

试题（26）、（27）

设系统中有 R 类资源 m 个，现有 n 个进程互斥使用。若每个进程对 R 资源的最大需求为 w，那么当 m、n、w 取下表的值时，对于下表中的 a～e 五种情况，__（26）__两种情况可能会发生死锁。对于这两种情况，若将__（27）__，则不会发生死锁。

	a	b	c	d	e
m	2	2	2	4	4
n	1	2	2	3	3
w	2	1	2	2	3

（26）A. a 和 b　　B. b 和 c　　C. c 和 d　　D. c 和 e

（27）A. n 加 1 或 w 加 1　　B. m 加 1 或 w 减 1

C. m 减 1 或 w 加 1　　D. m 减 1 或 w 减 1

试题（26）、（27）分析

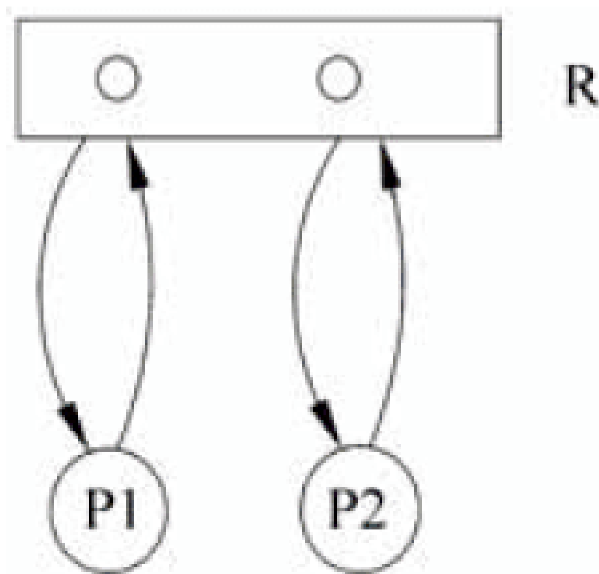
本题考查对操作系统死锁方面基本知识掌握的程度。系统中同类资源分配不当会引起死锁。一般情况下，若系统中有 m 个单位的存储器资源，它被 n 个进程使用，当每个进程都要求 w 个单位的存储器资源，当 m<nw 时，可能会引起死锁。

试题（26）分析如下：

情况 a：m=2，n=1，w=2，系统中有两个资源，1 个进程使用，该进程最多要求两个资源，所以不会发生死锁。

情况 b：m=2，n=2，w=1，系统中有两个资源，两个进程使用，每个进程最多要求 1 个资源，所以不会发生死锁。

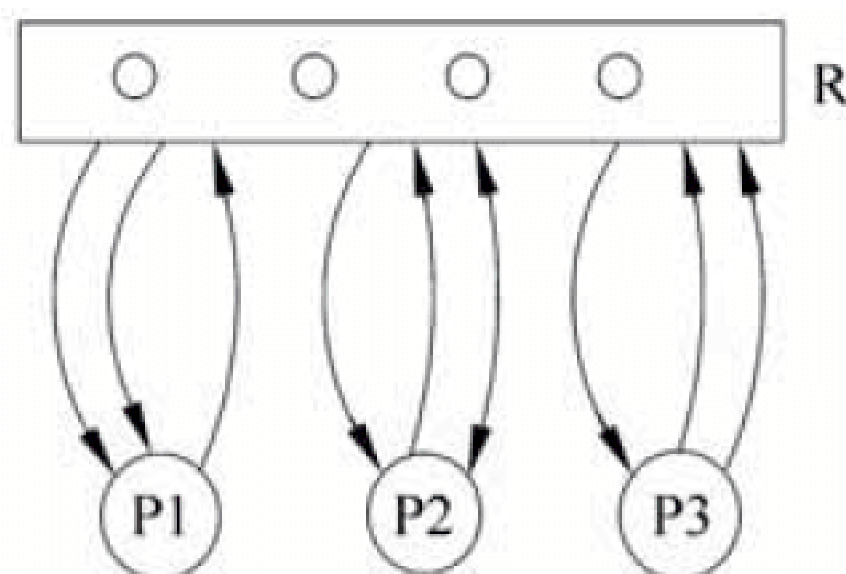
情况 c：m=2，n=2，w=2，系统中有两个资源，两个进程使用，每个进程最多要求两个资源，此时，采用的分配策略是轮流地为每个进程分配，则第一轮系统先为每个进程分配 1 个，此时，系统中已无可供分配的资源，使得各个进程都处于等待状态导致系统发生死锁，这时进程资源图如下图所示。



情况 d： m=4，n=3，w=2，系统中有 4 个资源，3 个进程使用，每个进程最多要求

两个资源，此时，采用的分配策略是轮流地为每个进程分配，则第一轮系统先为每个进程分配 1 个资源，此时，系统中还剩 1 个资源，可以使其中的一个进程得到所需资源运行完毕，所以不会发生死锁。

情况 e：m=4，n=3，w=3，系统中有 4 个资源，3 个进程使用，每个进程最多要求 3 个资源，此时，采用的分配策略是轮流地为每个进程分配，则第一轮系统先为每个进程分配 1 个，第二轮系统先为一个进程分配 1 个，此时，系统中已无可供分配的资源，使得各个进程都处于等待状态导致系统发生死锁，这时进程资源图如下图所示。



试题（27）分析如下：

对于 c 和 e 两种情况，若将 m 加 1，则情况 c： m=3，n=2，w=2，系统中有 3 个资源，两个进程使用，每个进程最多要求两个资源，系统先为每个进程分配 1 个，此时，系统中还剩 1 个可供分配的资源，使得其中的一个进程能得到所需资源执行完，并释放所有资源使另一个进程运行完；若将 w 减 1，则情况 c： m=2，n=2，w=1，系统中有两个资源，两个进程各需一个，系统为每个进程分配 1 个，此时，进程都能运行完，显然不会发生死锁。情况 e 分析同理。

参考答案

（26）D　（27）B

试题（28）

某文件系统采用链式存储管理方案，磁盘块的大小为 1024 字节。文件 Myfile.doc 由 5 个逻辑记录组成，每个逻辑记录的大小与磁盘块的大小相等，并依次存放在 121、75、86、65 和 114 号磁盘块上。若需要存取文件的第 5120 字节处的信息，应该访问<u>（28）</u>号磁盘块。

（28）A. 75　　B. 85　　C. 65　　D. 114

试题（28）分析

本题考查对操作系统文件系统空间管理方面基本知识掌握的程度。

根据题意每个逻辑记录的大小与磁盘块大小相等，并依次存放在 121、75、86、65 和 114 号磁盘块上。而文件的第 5120 字节应该在 114 号磁盘块上。

参考答案

（28）D

试题（29）

软件能力成熟度模型（CMM）将软件能力成熟度自低到高依次划分为 5 级。目前，达到 CMM 第 3 级（已定义级）是许多组织努力的目标，该级的核心是<u>（29）</u>。

（29）A. 建立基本的项目管理和实践来跟踪项目费用、进度和功能特性
B. 使用标准开发过程（或方法论）构建（或集成）系统
C. 管理层寻求更主动地应对系统的开发问题
D. 连续地监督和改进标准化的系统开发过程

试题（29）分析

本题考查软件成熟度模型（CMM）的基本概念。建立基本的项目管理和实践来跟踪项目费用、进度和功能特性为可重复级的核心；使用标准开发过程（或方法论）构建（或集成）系统为已定义级的核心；管理层寻求更主动地应对系统的开发问题为已管理级的核心；连续地监督和改进标准化的系统开发过程为优化级的核心。

参考答案

（29）B

试题（30）

RUP 在每个阶段都有主要目标，并在结束时产生一些制品。在<u>（30）</u>结束时产生"在适当的平台上集成的软件产品"。

（30）A. 初启阶段　　B. 精化阶段　　C. 构建阶段　　D. 移交阶段

试题（30）分析

本题考查 RUP 中每个阶段产生的制品。初启阶段结束时产生一个构想文档、一个有关用例模型的调查、一个初始的业务用例、一个早期的风险评估和一个可以显示阶段和迭代的项目计划等制品；精化阶段结束时产生一个补充需求分析、一个软件架构描述和一个可执行的架构原型等制品；构建阶段结束时的成果是一个准备交到最终用户手中的产品，包括具有最初运作能力的在适当的平台上集成的软件产品、用户手册和对当前版本的描述；移交阶段结束时产生移交给用户产品发布版本。

参考答案

（30）C

试题（31）

根据 ISO/IEC 9126 软件质量度量模型定义，一个软件的时间和资源质量子特性属于<u>（31）</u>质量特性。

（31）A. 功能性　　B. 效率　　C. 可靠性　　D. 易使用性

试题（31）分析

本题考查 ISO/IEC 9126 软件质量度量模型中的质量特性。效率质量特性包括时间特

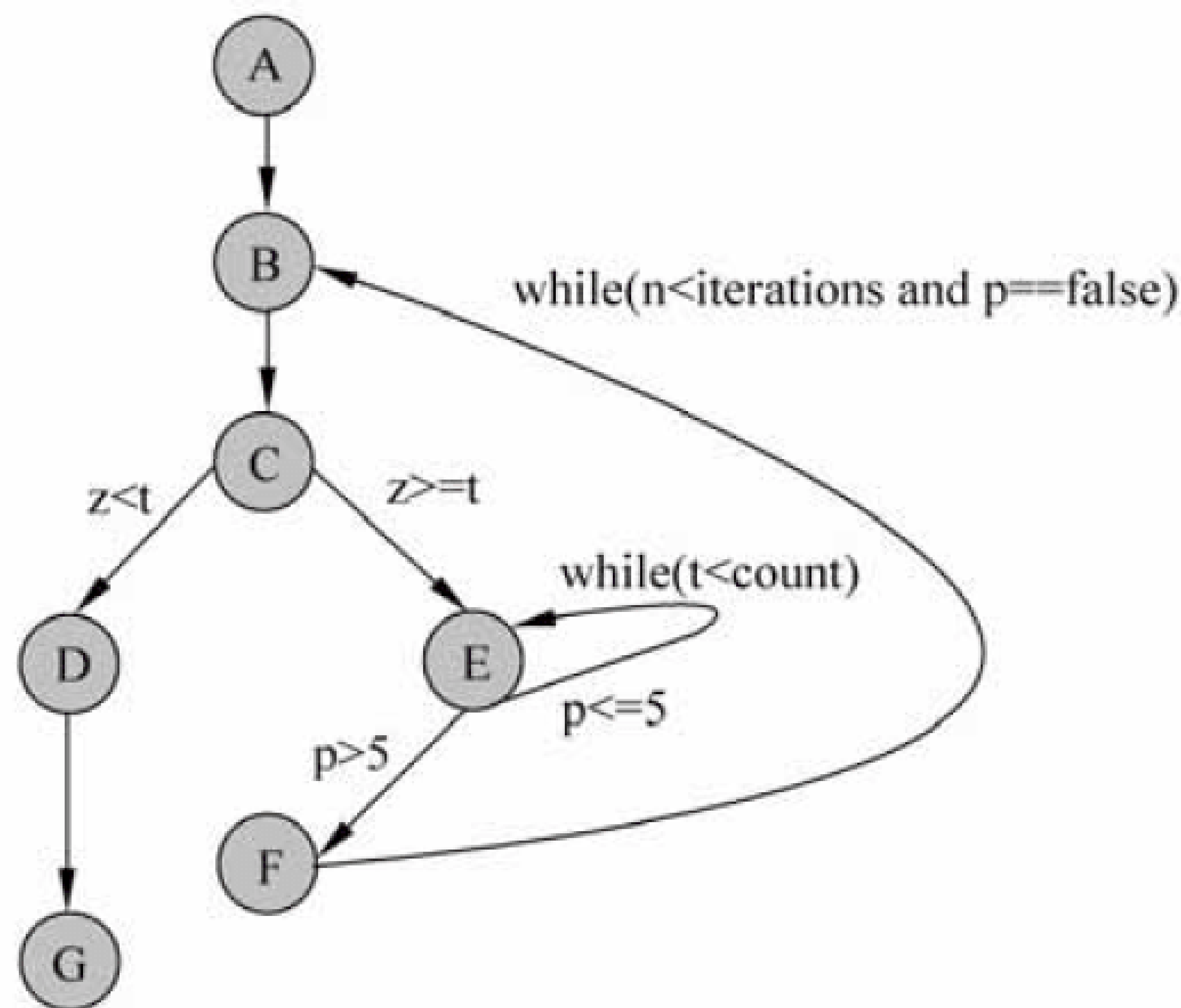
性和资源特性两个质量子特性。

参考答案

（31）B

试题（32）

McCabe 度量法是通过定义环路复杂度，建立程序复杂性的度量，它基于一个程序模块的程序图中环路的个数。计算有向图 G 的环路复杂性的公式为：$V(G)=m-n+2$，其中 $V(G)$是有向图 G 中的环路个数，m 是 G 中的有向弧数，n 是 G 中的节点数。下图所示程序图的程序复杂度是__（32）__。



（32）A. 2　　B. 3　　C. 4　　D. 5

试题（32）分析

本题考查 McCabe 度量法。要采用 McCabe 度量法度量程序复杂度，需要先画出程序流图，识别有向图中节点数 n=8，有向弧数 m=7，然后利用环路复杂性计算公式 $V(G)=m-n+2$ 进行计算，得如图所示的程序图的程序复杂度是 3。

参考答案

（32）B

试题（33）

在开发信息系统时，用于系统开发人员与项目管理人员沟通的主要文档是__（33）__。

（33）A. 系统开发合同　　B. 系统设计说明书
C. 系统开发计划　　D. 系统测试报告

试题（33）分析

本题考查开发文档的作用。系统开发人员与项目管理人员在项目期内进行沟通的文档主要有系统开发计划、系统开发月报以及系统开发总结报告等项目管理文件。

参考答案

（33）C

试题（34）

软件工程每一个阶段结束前，应该着重对可维护性进行复审。在系统设计阶段的复审期间，应该从__（34）__出发，评价软件的结构和过程。

（34）A. 指出可移植性问题以及可能影响软件维护的系统界面

B. 容易修改、模块化和功能独立的目的

C. 强调编码风格和内部说明文档

D. 可测试性

试题（34）分析

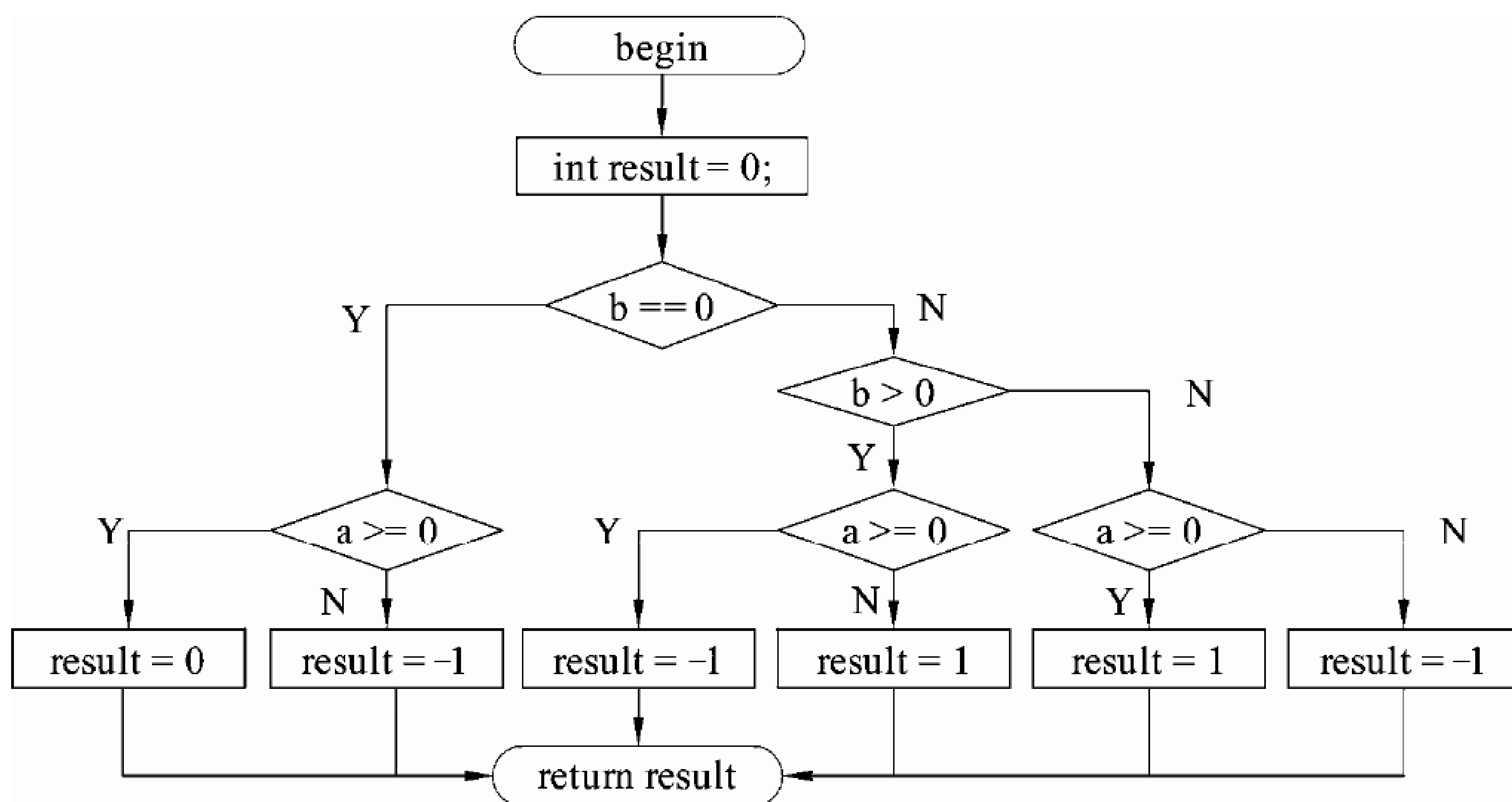
本题考查软件复审基本概念。可维护性是所有软件都应具有的基本特点，必须在开发阶段保证软件具有可维护的特点。在系统分析阶段的复审过程中，应该指出软件的可移植性问题以及可能影响软件维护的系统界面；在系统设计阶段的复审期间，应该从容易修改、模块化和功能独立的目的出发，评价软件的结构和过程；在系统实施阶段的复审期间，代码复审应该强调编码风格和内部说明文档这两个影响可维护性的因素。可测试性是可维护性的一个评价指标。

参考答案

（34）B

试题（35）

当用分支覆盖法对以下流程图进行测试时，至少需要设计__（35）__个测试用例。



（35）A. 4　　B. 5　　C. 6　　D. 8

试题（35）分析

本题考查分支覆盖测试法。采用分支覆盖法进行测试使得被测程序中每个判定表达

式至少获得一次“真”值和“假”值，或者程序中的每一个取“真”分支和取“假”分支至少都通过一次。需要设计的测试用例需要确保每一个取“真”分支和取“假”分支至少都通过一次，即统计最多分支数。

参考答案

（35）C

试题（36）

某银行为了使其网上银行系统能够支持信用卡多币种付款功能而进行扩充升级，这需要对数据类型稍微进行一些改变，这一状况需要对网上银行系统进行__（36）__维护。

（36）A. 正确性　　B. 适应性　　C. 完善性　　D. 预防性

试题（36）分析

本题考查软件维护的内容。为了使网上银行系统能够支持信用卡的多币种付款功能而扩充升级，是为了适应银行的市场环境和新的管理需求而提出的新的信息需求，因此需要适应性维护。

参考答案

（36）B

试题（37）

下面关于面向对象分析与面向对象设计的说法中，不正确的是__（37）__。

（37）A. 面向对象分析侧重于理解问题
B. 面向对象设计侧重于理解解决方案
C. 面向对象分析描述软件要做什么
D. 面向对象设计一般不关注技术和实现层面的细节

试题（37）分析

本题考查面向对象分析与设计的基本概念。

面向对象分析主要强调理解问题是什么，不考虑问题的解决方案，因此答案 A、C 是正确的。面向对象设计侧重问题的解决方案，并且需要考虑实现细节问题，因此选项 D 的说法是不正确的。

参考答案

（37）D

试题（38）～（40）

在面向对象分析与设计中，__（38）__是应用领域中的核心类，一般用于保存系统中的信息以及提供针对这些信息的相关处理行为；__（39）__是系统内对象和系统外参与者的联系媒介；__（40）__主要是协调上述两种类对象之间的交互。

（38）A. 控制类　　B. 边界类　　C. 实体类　　D. 软件类
（39）A. 控制类　　B. 边界类　　C. 实体类　　D. 软件类
（40）A. 控制类　　B. 边界类　　C. 实体类　　D. 软件类

试题（38）～（40）分析

本题考查控制类、边界类和实体类的职责。

实体类主要负责数据和业务逻辑；边界类负责和用户进行交互，即用户界面；控制类则负责实体类和界面类的交互。

参考答案

（38）C　（39）B　（40）A

试题（41）、（42）

若类 A 仅在其方法 Method1 中定义并使用了类 B 的一个对象，类 A 其他部分的代码都不涉及类 B，那么类 A 与类 B 的关系应为__（41）__；若类 A 的某个属性是类 B 的一个对象，并且类 A 对象消失时，类 B 对象也随之消失，则类 A 与类 B 的关系应为__（42）__。

（41）A. 关联　B. 依赖　C. 聚合　D. 组合

（42）A. 关联　B. 依赖　C. 聚合　D. 组合

试题（41）、（42）分析

本题考查类间的关系。

类间关系可分为依赖、关联、聚合、组合和继承 5 种。按照上述顺序，类间关系依次增强，若类 A 的方法中仅仅使用了类 B 的对象，那么类 A 依赖于类 B。如果类 A 的部分是由类 B 的对象组成，并且类 A 控制类 B 的生命周期，那么类 A 与类 B 是组合关系。

参考答案

（41）B　（42）D

试题（43）～（45）

当不适合采用生成子类的方法对已有的类进行扩充时，可以采用__（43）__设计模式动态地给一个对象添加一些额外的职责；当应用程序由于使用大量的对象，造成很大的存储开销时，可以采用__（44）__设计模式运用共享技术来有效地支持大量细粒度的对象；当想使用一个已经存在的类，但其接口不符合需求时，可以采用__（45）__设计模式将该类的接口转换成我们希望的接口。

（43）A. 命令（Command）　B. 适配器（Adapter）
C. 装饰（Decorate）　D. 享元（Flyweight）

（44）A. 命令（Command）　B. 适配器（Adapter）
C. 装饰（Decorate）　D. 享元（Flyweight）

（45）A. 命令（Command）　B. 适配器（Adapter）
C. 装饰（Decorate）　D. 享元（Flyweight）

试题（43）～（45）分析

本题考查面向对象设计。

装饰模式主要的目的是在无法生成子类的情况下给一个对象动态地增加新的职责；

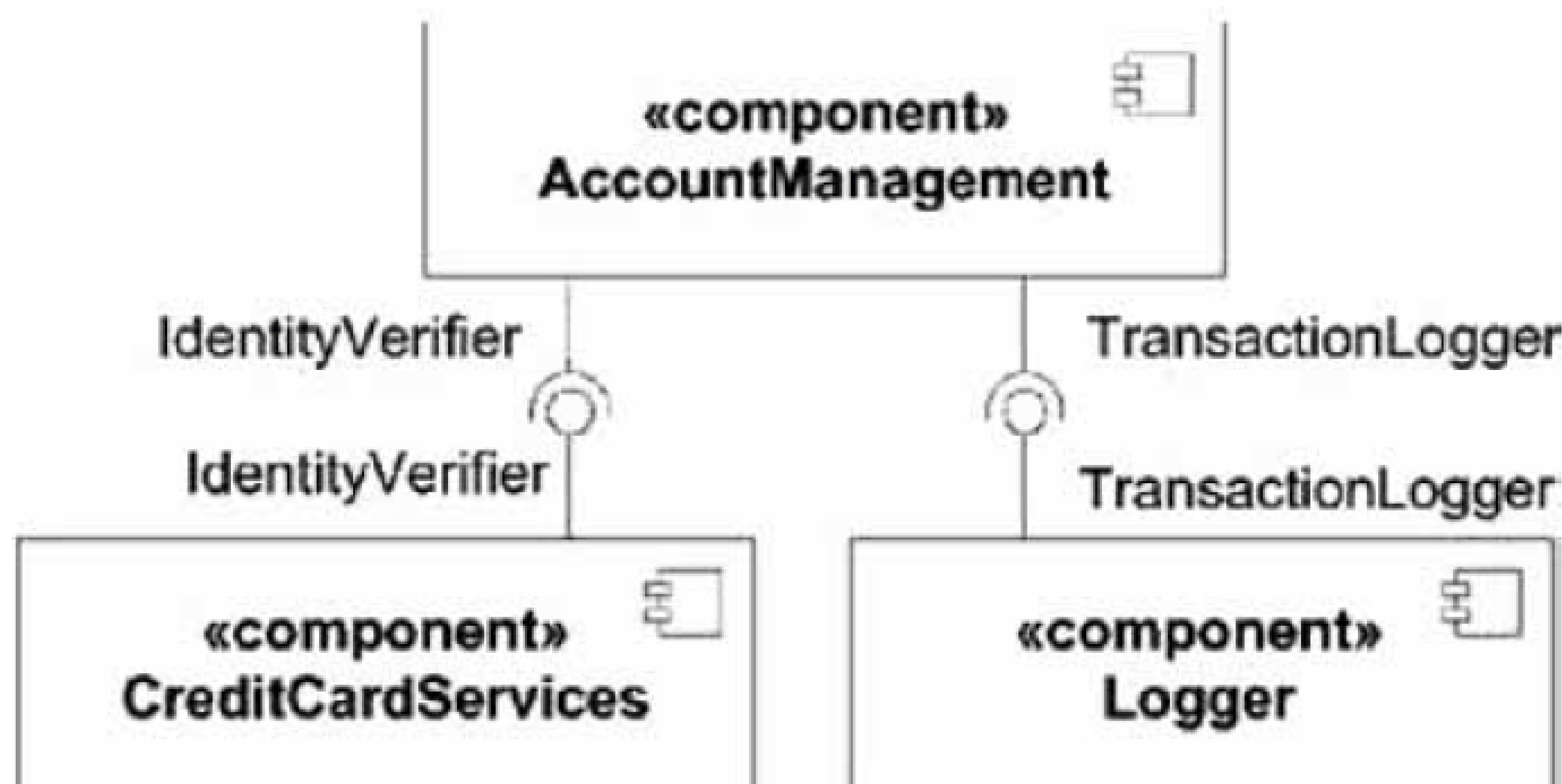
享元设计模式是共享大量细粒度的对象；适配器设计模式则是将已有的接口转换为系统希望的接口形式。

参考答案

（43）C　（44）D　（45）B

试题（46）、（47）

下图属于UML中的__（46）__，其中，AccountManagement需要__（47）__。



（46）A. 组件图　　B. 部署图　　C. 类图　　D.对象图

（47）A. 实现IdentityVerifier接口并被CreditCardServices调用

B. 调用CreditCardServices实现的IdentityVerifier接口

C. 实现IdentityVerifier接口并被Logger调用

D. 调用Logger实现的IdentityVerifier接口

试题（46）、（47）分析

本题考查UML语言基础知识。

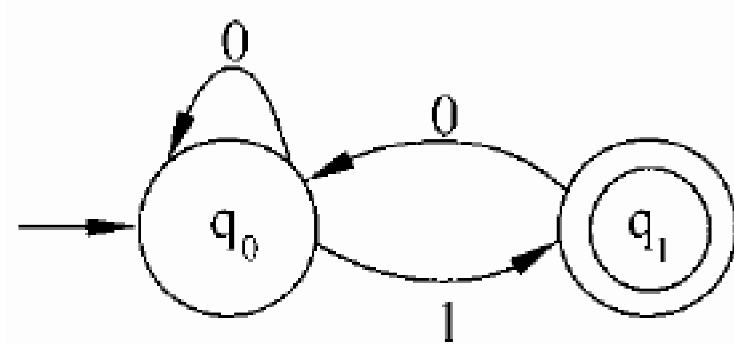
UML语言是标准的建模语言，通过图形化的方式展现系统的模型。本题是UML中的组件图，其表示的含义是组件AccountManagement需要调用CreditCardServices组件和Logger组件分别实现的IdentityVerifier接口和TransactionLogger接口。

参考答案

（46）A　（47）B

试题（48）

下图所示有限自动机的特点是__（48）__。



（48）A. 识别的0、1串是以0开头且以1结尾

B. 识别的0、1串中1的数目为偶数

C. 识别的 0、1 串中 0 后面必须是 1

D. 识别的 0、1 串中 1 不能连续出现

试题（48）分析

本题考查程序语言方面的基础知识。

对于题中自动机的状态图，先忽略状态 q0 的自环（识别若干个 0），从初态 q0 到终态 q1，该自动机可识别的字符串为 1、101、10101、…，显然，该自动机识别的 0、1 串中 1 不能连续出现。

参考答案

（48）D

试题（49）

由 a、b 构造且仅包含偶数个 a 的串的集合用正规式表示为 （49） 。

（49）A. (a*a)*b*　　B. (b* (ab*a)*)*

C. (a* (ba*)*b)*　　D. (a|b)* (aa)*

试题（49）分析

本题考查程序语言方面的基础知识。

在正规式中，符号*表示重复若干次（包括 0 次），因此正规式“ (a*a)*b*”中的表达式“(a*a) *”不能保证有偶数个 a。同理，“(a* (ba*)*b)*”和“(a|b)* (aa)*”中对 a 的个数也没有限制，而在“(ab*a)*”中可以确保 a 的出现为偶数个。

参考答案

（49）B

试题（50）

设某语言的语法规则用上下文无关文法 G=(N,T,P,S)表示，其中 N 是非终结符号的集合，T 是终结符号的集合，P 是产生式集合，S 是开始符号，令 V=N∪T，那么符合该语言的句子是 （50） 。

（50）A. 从 S 出发推导的、仅包含 T 中符号的符号串

B. 从 N 中符号出发推导的、仅包含 T 中符号的符号串

C. 从 S 出发推导的、包含 V 中符号的符号串

D. 从 N 中符号出发推导的、包含 V 中符号的符号串

试题（50）分析

本题考查程序语言方面的基础知识。

一个文法的语言是该文法能产生的句子的集合。一个文法产生的句子是从文法开始符号出发推导出的所有终结符号串。

参考答案

（50）A

试题（51）

采用二维表格结构表达实体类型及实体间联系的数据模型是__（51）__。

（51）A. 层次模型　　B. 网状模型

C. 关系模型　　D. 面向对象模型

试题（51）分析

本题考查对数据库数据模型方面基本知识掌握的程度。

不同的数据模型具有不同的数据结构形式。目前最常用的数据结构模型有层次模型（hierarchical model）、网状模型（network model）、关系模型（relational Model）和面向对象数据模型（object oriented model）。其中层次模型和网状模型统称为非关系模型。非关系模型的数据库系统在20世纪70年代非常流行，在数据库系统产品中占据了主导地位。到了20世纪80年代，逐渐被关系模型的数据库系统取代，但某些地方，由于历史的原因，目前层次和网状数据库系统仍在使用。

关系模型是目前最常用的数据模型之一。关系数据库系统采用关系模型作为数据的组织方式，在关系模型中用二维表格结构表达实体集以及实体集之间的联系，其最大特色是描述的一致性。关系模型是由若干个关系模式组成的集合。一个关系模式相当于一个记录型，对应于程序设计语言中类型定义的概念。关系是一个实例，也是一张表，对应于程序设计语言中变量的概念。给定变量的值随时间可能发生变化；类似地，当关系被更新时，关系实例的内容也随时间发生了变化。

参考答案

（51）C

试题（52）～（54）

假设员工关系EMP（员工号，姓名，部门，部门电话，部门负责人，家庭住址，家庭成员，成员关系）如下表所示。如果一个部门可以有多名员工，一个员工可以有多个家庭成员，那么关系EMP属于__（52）__，且__（53）__问题；为了解决这一问题，应该将员工关系EMP分解为__（54）__。

员工号	姓名	部门	部门电话	部门负责人	家庭住址	家庭成员	成员关系
0011	张晓明	开发部	808356	0012	北京海淀区1号	张大军	父亲
0011	张晓明	开发部	808356	0012	北京海淀区1号	胡敏铮	母亲
0011	张晓明	开发部	808356	0012	北京海淀区1号	张晓丽	妹妹
0012	吴俊	开发部	808356	0012	上海昆明路15号	吴胜利	父亲
0012	吴俊	开发部	808356	0012	上海昆明路15号	王若垚	母亲
0021	李立丽	市场部	808358	0021	西安雁塔路8号	李国庆	父亲
0021	李立丽	市场部	808358	0021	西安雁塔路8号	罗明	母亲
0022	王学强	市场部	808356	0021	西安太白路2号	王国钧	父亲
0031	吴俊	财务部	808360		西安科技路18号	吴鸿翔	父亲

（52）A. 1NF　　B. 2NF　　C. 3NF　　D. BCNF

（53）A. 无冗余、无插入异常和删除异常

B. 无冗余，但存在插入异常和删除异常

C. 存在冗余，但不存在修改操作的不一致

D. 存在冗余、修改操作的不一致，以及插入异常和删除异常

（54）A. EMP1（员工号，姓名，家庭住址）

EMP2（部门，部门电话，部门负责人）

EMP3（员工号，家庭成员，成员关系）

B. EMP1（员工号，姓名，部门，家庭住址）

EMP2（部门，部门电话，部门负责人）

EMP3（员工号，家庭成员，成员关系）

C. EMP1（员工号，姓名，家庭住址）

EMP2（部门，部门电话，部门负责人，家庭成员，成员关系）

D. EMP1（员工号，姓名，部门，部门电话，部门负责人，家庭住址）

EMP2（员工号，家庭住址，家庭成员，成员关系）

试题（52）～（54）分析

本题考查应试者对范式、模式分解知识的掌握程度。

试题（52）考查的是范式的基础知识。员工关系 EMP 属于第一范式的原因是因为其主键是（员工号，家庭成员），非主属性部门名，负责人，电话存在对主键的部分函数依赖。所以正确的答案是 A。

试题（53）正确的答案是 D，因为表中存在冗余、修改操作的不一致，以及插入异常和删除异常。

试题（54）正确的答案是 B，因为对一个给定的关系模式进行分解，使得分解后的模式是否与原来的模式等价有如下三种情况：

① 分解具有无损连接性；

② 分解要保持函数依赖；

③ 分解既要无损连接性，又要保持函数依赖。

选项 A 是错误的，因为将原关系模式分解成 EMP1（员工号，姓名，家庭住址），EMP2（部门，部门电话，部门负责人）和 EMP3（员工号，家庭成员，成员关系）三个关系模式，分解后的关系模式既是有损连接，又不能保持函数依赖。因为此时给定员工号已无法查找所在的部门，如下表所示。

EMP1

员工号	姓名	家庭住址
0011	张晓明	北京海淀区 1 号
0012	吴俊	上海昆明路 15 号
0021	李立丽	西安雁塔路 8 号
0022	王学强	西安太白路 2 号
0031	吴俊	西安科技路 18 号

EMP3

部门	部门电话	部门负责人
开发部	808356	0012
市场部	808358	0021
财务部	808360	

EMP2

员工号	家庭成员	成员关系
0011	张大军	父亲
0011	胡敏铮	母亲
0011	张晓丽	妹妹
0012	吴胜利	父亲
0012	王若垚	母亲
0021	李国庆	父亲
0021	罗明	母亲
0022	王国钧	父亲
0031	吴鸿翔	父亲

选项 B 是正确的，因为将原关系模式分解成 EMP1（员工号，姓名，部门，家庭住址）、EMP2（部门，部门电话，部门负责人）和 EMP3（员工号，家庭成员，成员关系）既具有无损连接性，又保持了函数依赖。如下表所示。

EMP1

员工号	姓名	部门	家庭住址
0011	张晓明	开发部	北京海淀区 1 号
0012	吴俊	开发部	上海昆明路 15 号
0021	李立丽	市场部	西安雁塔路 8 号
0022	王学强	市场部	西安太白路 2 号
0031	吴俊	财务部	西安科技路 18 号

EMP3

部门	部门电话	部门负责人
开发部	808356	0012
市场部	808358	0021
财务部	808360	

EMP2

员工号	家庭成员	成员关系
0011	张大军	父亲
0011	胡敏铮	母亲
0011	张晓丽	妹妹
0012	吴胜利	父亲
0012	王若垚	母亲
0021	李国庆	父亲
0021	罗明	母亲
0022	王国钧	父亲
0031	吴鸿翔	父亲

选项 C 是错误的，因为将原关系模式分解成 EMP1（员工号，姓名，家庭住址）和 EMP2（部门，部门电话，部门负责人，家庭成员，成员关系）两个关系模式，分解后的关系模式既是有损连接，又不能保持函数依赖。例如，给定员工号无法查找所在的部门，无法查找其家庭成员等信息。如下表所示。

EMP1

员工号	姓名	家庭住址
0011	张晓明	北京海淀区 1 号
0012	吴俊	上海昆明路 15 号
0021	李立丽	西安雁塔路 8 号
0022	王学强	西安太白路 2 号
0031	吴俊	西安科技路 18 号

EMP2

部门	部门电话	部门负责人	家庭成员	成员关系
开发部	808356	0012	张大军	父亲
开发部	808356	0012	胡敏铮	母亲
开发部	808356	0012	张晓丽	妹妹
开发部	808356	0012	吴胜利	父亲
开发部	808356	0012	王若垚	母亲
市场部	808358	0021	李国庆	父亲
市场部	808358	0021	罗明	母亲
市场部	808356	0021	王国钧	父亲
财务部	808360		吴鸿翔	父亲

选项 D 是错误的，因为将原关系模式分解成 EMP1（员工号，姓名，部门，部门电话，部门负责人，家庭住址）和 EMP2（员工号，家庭住址，家庭成员，成员关系）两个关系模式，分解后的关系模式存在冗余和修改操作的不一致性。例如，EMP1 中某员工的家庭住址从“陕西省西安市太白路 2 号”修改为“陕西省西安市雁塔路 18 号”，而 EMP2 中该员工的家庭住址未修改，导致修改操作的不一致性。又如，EMP2 中某员工的家庭成员有 5 个，那么其家庭住址就要重复出现 5 次，导致数据的冗余。

参考答案

（52）A　（53）D　（54）B

试题（55）、（56）

关系 R、S 如下图所示，关系代数表达式 $\pi_{3,4,5}(\sigma_{1<6}(R\times S))$ = ___（55）___，对关系 R、S 进行自然连接后的属性列数和元组个数分别为___（56）___。

A	B	C
1	2	4
3	4	5
4	5	9
5	6	6

R

A	B	C
5	3	3
4	6	1
9	8	3
6	9	1

S

（55）A.

A	B	C
1	2	4
5	3	3

B.

A	B	C
5	3	4
9	8	4

C.

A	B	C
5	3	3
9	8	3

D.

A	B	C
1	2	4
3	4	5

（56）A. 3 和 0　　B. 3 和 2　　C. 6 和 0　　D. 6 和 2

试题（55）、（56）分析

本题考查对关系代数运算方面的基础知识。

本题要求关系代数表达式 $\pi_{3,4,5}(\sigma_{1<6}(R\times S))$ 的结果集，其中，R×S 的属性列名分别为 R..A， R.B， R.C， S.A， S.B 和 S.C，其结果如下表所示：

R.A	R.B	R.C	S.A	S.B	S.C
1	2	4	5	3	3
1	2	4	4	6	1
1	2	4	9	8	3
1	2	4	6	9	1
3	4	5	5	3	3
3	4	5	4	6	1
3	4	5	9	8	3
3	4	5	6	9	1
4	5	9	5	3	3
4	5	9	4	6	1
4	5	9	9	8	3
4	5	9	6	9	1
5	6	6	5	3	3
5	6	6	4	6	1
5	6	6	9	8	3
5	6	6	6	9	1

R×S

$\sigma_{1<6}(R\times S)$ 的含义是从 R×S 结果集中选取第一个分量（R.A）小于第 6 个分量（S.C）的元组，从上表中可以看出，满足条件的是第一和第三个元组，其结果如下表所示：

R.A	R.B	R.C	S.A	S.B	S.C
1	2	4	5	3	3
1	2	4	9	8	3

$\sigma_{1<6}(R\times S)$

$\pi_{3,4,5}(\sigma_{1<6}(R\times S))$ 的含义是从 $\sigma_{1<6}(R\times S)$ 结果集中选取第三列、第四列和第五列，其结果如图（a）所示；表中 S.A、S.B 和 R.C 的属性名不重复，所以可以用图（b）表示。

S.A	S.B	R.C
5	3	4
9	8	4

$\pi_{3,4,5}(\sigma_{1<6}(R\times S))$

图（a）

A	B	C
5	3	4
9	8	4

$\pi_{3,4,5}(\sigma_{1<6}(R\times S))$

图（b）

从上面分析可见，试题（55）的正确答案是 B。

试题（56）的正确答案是 A，因为根据自然连接要求，两个关系中进行比较的分量必须是相同的属性组，并且在结果中将重复属性列去掉，故 $R \bowtie S$ 后的属性列数应为 3；又因为自然连接是一种特殊的等值连接，即 R 关系中的 A、C、D 属性与 S 关系中的 A、C、D 属性进行等值连接没有符合条件的元组，故其结果集的元组个数为 0。

参考答案

（55）B　　（56）A

试题（57）

下面关于查找运算及查找表的叙述，错误的是＿（57）＿。

（57）A. 哈希表可以动态创建

B. 二叉排序树属于动态查找表

C. 二分查找要求查找表采用顺序存储结构或循环链表结构

D. 顺序查找方法既适用于顺序存储结构，也适用于链表结构

试题（57）分析

本题考查数据结构方面的基础知识。

哈希表和二叉排序树都可以在查找过程中动态创建，属于动态查找表。顺序查找方法按照设定的次序依次与查找表中元素的关键字进行比较，在顺序存储结构和链表结构上都可以实现该查找过程。二分查找需要对中间元素进行快速定位，在链表结构上无法实现。

参考答案

（57）C

试题（58）

下面关于图（网）的叙述，正确的是＿（58）＿。

（58）A. 连通无向网的最小生成树中，顶点数恰好比边数多 1

B. 若有向图是强连通的，则其边数至少是顶点数的 2 倍

C. 可以采用 AOV 网估算工程的工期

D. 关键路径是 AOE 网中源点至汇点的最短路径

试题（58）分析

本题考查数据结构方面的基础知识。

在有向图中，若以顶点表示活动，用有向边表示活动之间的优先关系，则称这样的有向图为以顶点表示活动的网（Activity On Vertex Network，AOV）。

若在带权有向图 G 中以顶点表示事件，以有向边表示活动，边上的权值表示该活动持续的时间，则这种带权有向图称为用边表示活动的网（Activity On Edge Network，AOE 网）通常在 AOE 网中列出了完成预定工程计划所需进行的活动、每项活动的计划完成时间、要发生哪些事件以及这些事件和活动间的关系，从而可以分析该项工程是否实际

可行并估计工程完成的最短时间，分析出哪些活动是影响工程进度的关键。进一步可以进行人力、物力的调度和分配，以达到缩短工期的目的。

根据生成树的定义，有 n 个顶点的连通图的生成树中恰好有 n–1 条边。

参考答案

（58）A

试题（59）

下面关于二叉排序树的叙述，错误的是__（59）__。

（59）A. 对二叉排序树进行中序遍历，必定得到节点关键字的有序序列

B. 依据关键字无序的序列建立二叉排序树，也可能构造出单支树

C. 若构造二叉排序树时进行平衡化处理，则根节点的左子树节点数与右子树节点数的差值一定不超过 1

D. 若构造二叉排序树时进行平衡化处理，则根节点的左子树高度与右子树高度的差值一定不超过 1

试题（59）分析

本题考查数据结构方面的基础知识。

二叉排序树或者是一棵空树，或者是具有如下性质的二叉树：

① 若它的左子树非空，则其左子树上所有节点的关键字均小于根节点的关键字；

② 若它的右子树非空，则其右子树上所有节点的关键字均大于根节点的关键字；

③ 左、右子树本身就是两棵二叉排序树。

由上述定义可知，二叉排序树是一个有序表，对二叉排序树进行中序遍历，可得到一个关键字递增排序的序列。

对于给定的关键字序列，可从空树开始，逐个将关键字插入树中来构造一棵二叉排序树。其过程是：每读入一个关键字值，就建立一个新节点。若二叉排序树非空，则将新节点的关键字与根节点的关键字相比较，如果小于根节点的值，则插入到左子树中，否则插入到右子树中；若二叉排序树为空树，则新节点作为二叉排序树的根节点。

显然，若关键字初始序列已经有序，则构造出的二叉排序树一定是单枝树（每个节点只有一个孩子）。

为了使在二叉排序树上进行的查找操作性能最优，构造二叉排序树时需进行平衡化处理，使每个节点左、右子树的高度差的绝对值不超过 1。

参考答案

（59）C

试题（60）

下面关于栈和队列的叙述，错误的是__（60）__。

（60）A. 栈和队列都是操作受限的线性表

B. 队列采用单循环链表存储时，只需设置队尾指针就可使入队和出队操作的

时间复杂度都为 O(1)

C. 若队列的数据规模 n 可以确定，则采用顺序存储结构比链式存储结构效率更高

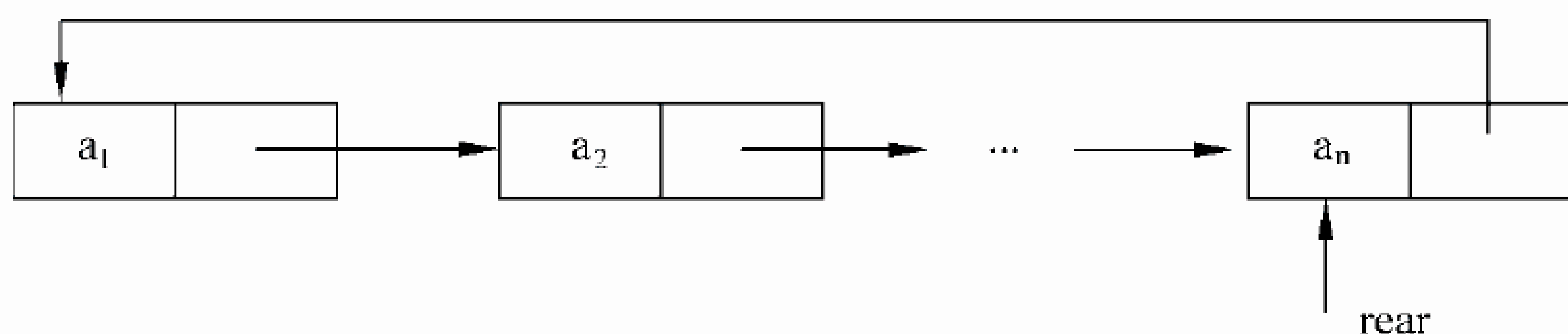
D. 利用两个栈可以模拟一个队列的操作，反之亦可

试题（60）分析

本题考查数据结构方面的基础知识。

栈和队列都是操作受限的线性表：栈仅在表尾插入和删除元素；队列仅在表头删除元素、在表尾插入元素。

采用单循环链表表示队列的示意图如下图所示：



① 入队时，新元素在 a_n 之后，若新元素节点指针为 s，则在一般情况下入队操作序列表示为 s->next=rear->next;　rear ->next=s;　rear=s;。

② 出队时，将队头元素 a_1 从队列中删除，一般情况下出队操作序列表示为：

```
q=rear->next;   //q 指向队头元素所在节点
 rear->next=q->next;
free(q);
```

入队时初始队列为空、出队后队列变为空要进行特殊处理。

入队操作和出队操作均与队列长度无关，因此其时间复杂度都为 O(1)。

队列是先入先出的线性表，栈是后进先出的线性表。一个线性序列经过队列结构后只能得到与原序列相同的元素序列，而经过一个栈结构后则可以得到多种元素序列。用两个栈可以模拟一个队列的入队和出队操作。

参考答案

（60）D

试题（61）

下面关于二叉树的叙述，正确的是__（61）__。

（61）A. 完全二叉树的高度 h 与其节点数 n 之间存在确定的关系

B. 在二叉树的顺序存储和链式存储结构中，完全二叉树更适合采用链式存储结构

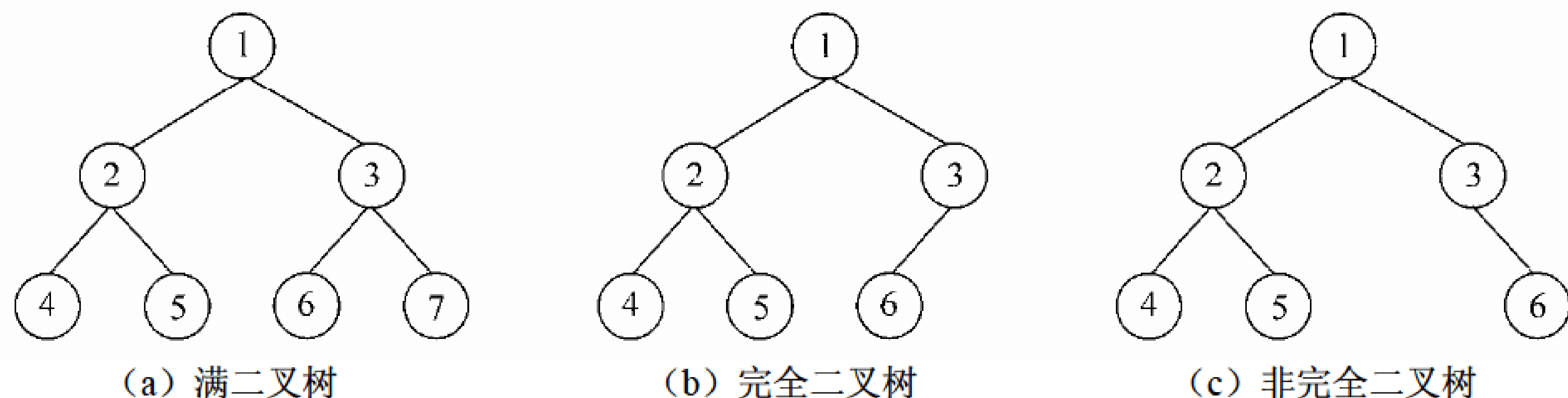
C. 完全二叉树中一定不存在度为 1 的节点

D. 完全二叉树中必定有偶数个叶子节点

试题（61）分析

本题考查数据结构方面的基础知识。

根据其定义，一棵完全二叉树除了最后一层外，其余层的节点数都是满的，最后一层的节点也必须自左至右排列，例如图（a）是高度为 3 的满二叉树，图（b）是完全二叉树，图（c）不是完全二叉树。

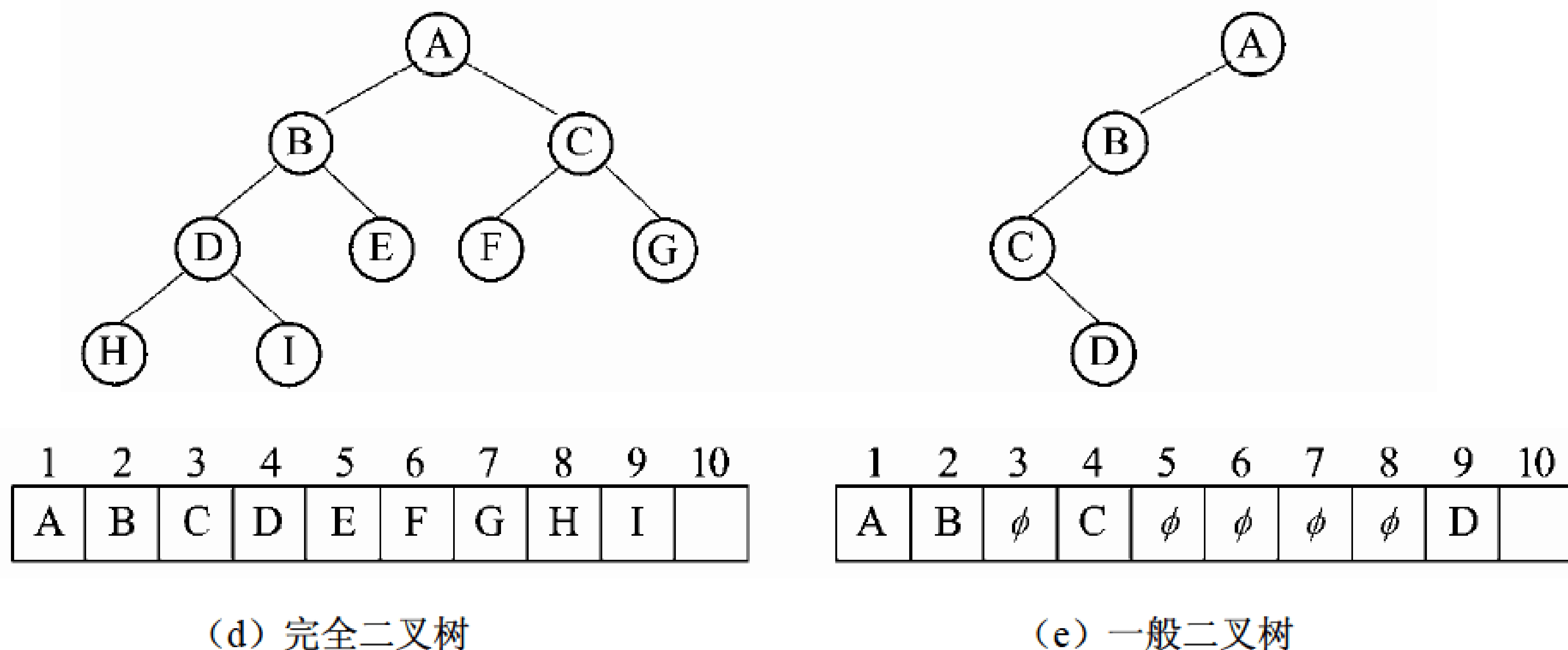


（a）满二叉树　　（b）完全二叉树　　（c）非完全二叉树

二叉树采用顺序存储结构时，对于编号为 i 的节点，则有：

- 若 i=1 时，该节点为根节点，无双亲；
- 若 i>1 时，该节点的双亲节点为$\lfloor i/2 \rfloor$；
- 若 2i≤n，则该节点的左孩子编号为 2i，否则无左孩子；
- 若 2i+1≤n，则该节点的右孩子编号为 2i+1，否则无右孩子。

图（d）为具有 10 个节点的完全二叉树及其顺序存储结构，图（e）为某非完全二叉树的顺序存储结构，从中可以看出，完全二叉树适合采用顺序存储结构。



1	2	3	4	5	6	7	8	9	10
A	B	C	D	E	F	G	H	I	

1	2	3	4	5	6	7	8	9	10
A	B	ϕ	C	ϕ	ϕ	ϕ	ϕ	D	

（d）完全二叉树　　（e）一般二叉树

可以推导出具有 n 个节点的完全二叉树的深度为$\lfloor \log_2 n \rfloor + 1$。

参考答案

（61）A

试题（62）

设 L 为广义表，将 head(L)定义为取非空广义表的第一个元素，tail(L)定义为取非空

广义表除第一个元素外剩余元素构成的广义表。若广义表 L=((x,y,z),a,(u,t,w))，则从 L 中取出原子项 y 的运算是 （62） 。

（62）A. head(tail(tail(L)))　　B. tail(head(head(L)))
C. head(tail(head(L)))　　D. tail(tail(head(L)))

试题（62）分析

本题考查数据结构方面的基础知识。

广义表是函数式语言中使用的一种数据结构。根据广义表取表头和取表尾的定义，对于广义表 L=((x,y,z),a,(u,t,w))，运算 head(L)=(x,y,z)，而 tail(L) =(a,(u,t,w))，因此原子项 y 应从 head(L)中取，对(x,y,z)取表头可得到原子项 x，因此从 L 中取出原子项 y 的运算为 head(tail(head(L)))。

参考答案

（62）C

试题（63）

现有 16 枚外形相同的硬币，其中有一枚比真币的重量轻的假币，若采用分治法找出这枚假币，至少比较 （63） 次才能够找出该假币。

（63）A. 3　　B. 4　　C. 5　　D. 6

试题（63）分析

本题考查算法基础知识。

用分治法找假币的过程为：先将 16 枚硬币对等分为 2 堆（各 8 枚）并比较其重量，假币在较轻的那一堆中；然后将 8 枚硬币对等分为 2 堆（各 4 枚）并比较其重量，假币在较轻的那一堆中；再将 4 枚硬币对等分为 2 堆（各 2 枚）并比较其重量，假币在较轻的那一堆中；最后比较两个硬币的重量，找出假币。因此，至少比较 4 次才能够找出该假币。

参考答案

（63）B

试题（64）

以下的算法设计方法中， （64） 以获取问题最优解为目标。

（64）A. 回溯方法　　B. 分治法　　C. 动态规划　　D. 递推

试题（64）分析

本题考查算法基础知识。

回溯法的实质是在包含问题的所有解的解空间树中，按照深度优先的策略，从根节点出发搜索解空间树。若进入某子节点的子树后没有找到解（或者需要找出全部解），则需要从子节点回退（回溯）至父节点，从而可以选择其他子节点进行搜索。回溯法有“通用的解题法”之称，用它可以系统地搜索一个问题的所有解或任一解。

分治与递归就像一对孪生兄弟，经常同时应用于算法设计之中。分治的思路是将一个难以直接解决的大问题分解成一些规模较小的相同问题，以便各个击破，分而治之。

如果规模为 n 的问题可分解成 k 个子问题，1<k≤n，这些子问题互相独立且与原问题相同。

动态规划算法与分治法类似，其基本思想也是将待求解问题分解成若干个子问题，先求解子问题，然后从这些子问题的解得到原问题的解。与分治法不同的是，适合于用动态规划法求解的问题，经分解得到的子问题往往不是独立的。若用分治法来解这类问题，则相同的子问题会被求解多次，以至于最后解决原问题需要耗费指数级时间。动态规划算法通常用于求解具有某种最优性质的问题。在这类问题中，可能会有许多可行解，每个解都对应于一个值，我们希望找到具有最优值（最大值或最小值）的那个解。

参考答案

（64）C

试题（65）

归并排序采用的算法设计方法属于__（65）__。

（65）A. 归纳法　　B. 分治法　　C. 贪心法　　D. 回溯方法

试题（65）分析

本题考查算法基础知识。

以 2-路归并排序为例进行说明。2-路归并是指将两个有序序列合并成一个有序序列，其基本过程为：从两个序列中各取一个元素，进行比较，输出较小的元素，从较小元素所在序列取下一个元素，与未输出的那个元素比较，输出较小者。依此类推，直到输出序列包含了两个初始有序序列的全部元素。

对于一个初始无序的序列，可以先将其等分为两个无序的子序列，对这两个子序列再次二分，重复该过程，直到分出的子序列中仅包含一个元素时（一个元素自然是有序的）为止，然后再反复进行 2-路归并的过程，最后完成排序。

参考答案

（65）B

试题（66）

一个 B 类网络的子网掩码为 255.255.224.0，则这个网络被划分成了__（66）__个子网。

（66）A. 2　　B. 4　　C. 6　　D. 8

试题（66）分析

子网掩码 255.255.224.0 的二进制表示为 11111111.11111111.11100000.00000000，比正常的 B 类子网掩码为 255.255.0.0 多出了 3 位“1”，所以把 B 类网络划分成了 8 个子网。

参考答案

（66）D

试题（67）

在 Windows 系统中设置默认路由的作用是　（67）　。

（67）A．当主机接收到一个访问请求时首先选择的路由

B．当没有其他路由可选时最后选择的路由

C．访问本地主机的路由

D．必须选择的路由

试题（67）分析

Windows Server 2003 的路由类型有 5 种，见下表。当 Windows 服务器收到一个 IP 数据包时，先查找主机路由，再查找网络路由（直连网络和远程网络），这些路由查找失败时，最后才查找默认路由。

路由类型

路 由 类 型	说　明
直连网络 ID（Directly attached network ID）	用于直接连接的网络，Interface（或 next hop）可以为空
远程网络 ID（Remote network ID）	用于不直接连接的网络，可以通过其他路由器到达这种网络 Interface 字段是本地路由器的 IP 地址
主机路由（Host route）	到达特定主机的路由，子网掩码为 255.255.255.255
默认路由（Default route）	无法找到确定路由时使用的路由，目标网络和网络掩码都是 0.0.0.0
持久路由（Persistent route）	利用 route add -p 命令添加的表项，每次初始化时，这种路由都会加入 Windows 的注册表中，同时加入路由表

参考答案

（67）B

试题（68）

HTML<body>元素中，　（68）　属性用于定义超链接被鼠标点击后所显示的颜色。

（68）A. alink　　B. background　　C. bgcolor　　D. vlink

试题（68）分析

本题考查的是 HTML 基本标签的使用。alink 用于设置正在被击中的链接的颜色。vlink 用于设置已使用的链接的颜色。background 用于设置背景图片的 URL。bgcolor 用于设置文档整体背景颜色。

参考答案

（68）D

试题（69）

HTML 中<tr >标记用于定义表格的__（69）__。

（69）A．行　　B．列　　C．单元格　　D．标题

试题（69）分析

本题考查 HTML 标记中 tr 标记的使用。<tr>标记用于定义表格中的一行。<col>标记用于定义表格中一个或多个列的属性值。<td>标记用于定义表格中的一个单元格。<title>标记用于定义文档标题。

参考答案

（69）A

试题（70）

以下不符合 XML 文档语法规范的是__（70）__。

（70）A．文档的第一行必须是 XML 文档声明

B．文档必须包含根元素

C．每个开始标记必须和结束标记配对使用

D．标记之间可以交叉嵌套

试题（70）分析

本题考查 XML 语法的基础知识。XML 文件的第一行必须是声明该文件是 XML 文件以及它所使用的 XML 规范版本。在文件的前面不能够有其他元素或者注释。所有的 XML 文档必须有一个根元素。XML 文档中的第一个元素就是根元素。所有 XML 文档都必须包含一个单独的标记来定义，所有其他元素都必须成对地在根元素中嵌套。XML 文档有且只能有一个根元素。所有的元素都可以有子元素，子元素必须正确地嵌套在父元素中。在 XML 中规定，所有标识必须成对出现，有一个开始标识，就必须有一个结束标识，否则将被视为错误。

参考答案

（70）D

试题（71）～（75）

For nearly ten years, the Unified Modeling Language (UML) has been the industry standard for visualizing, specifying, constructing, and documenting the __（71）__ of a software-intensive system. As the __（72）__ standard modeling language, the UML facilitates communication and reduces confusion among project __（73）__. The recent standardization of UML 2.0 has further extended the language's scope and viability. Its inherent expressiveness allows users to __（74）__ everything from enterprise information systems and distributed Web-based applications to real-time embedded systems.

The UML is not limited to modeling software. In fact, it is expressive enough to model __（75）__ systems, such as workflow in the legal system, the structure and behavior of a

patient healthcare system, software engineering in aircraft combat systems, and the design of hardware.

To understand the UML, you need to form a conceptual model of the language, and this requires learning three major elements: the UML's basic building blocks, the rules that dictate how those building blocks may be put together, and some common mechanisms that apply throughout the UML.

（71）A. classes　B. components　C. sequences　D. artifacts
（72）A. real　B. legal　C. de facto　D. illegal
（73）A. investors　B. developers　C. designers　D. stakeholders
（74）A. model　B. code　C. test　D. modify
（75）A. non-hardware　B. non-software　C. hardware　D. software

参考译文

近十年来，统一建模语言（UML）已经成为工业标准，它可用来可视化、规范化说明、构建以及文档化软件密集系统中的开发制品。作为事实上的工业标准，UML 能够方便项目相关人员的沟通并减少理解上的二义问题。UML 2.0 标准扩宽了该语言的应用范围，它所具有的表达能力能够让用户对企业信息系统、分布式 Web 系统和嵌入式实时系统进行建模。

UML 不仅能够对软件系统进行建模，实际上，它具有足够的能力去对法律系统中的工作流、病人监护系统中的结构和行为、飞行战斗系统和硬件系统进行建模。

为了理解 UML，需要具备该语言的概念模型，这需要学习三个主要元素：UML 的基本构造块，基本构造块的关系规则和应用这些构造块与规则的通用机制。

参考答案

（71）D　（72）C　（73）D　（74）A　（75）B

第 2 章　2009 上半年软件设计师下午试题分析与解答

试题一（共 **15** 分）

阅读下列说明，回答问题 1 和问题 2，将解答填入答题纸的对应栏内。

【说明】

假设某大型商业企业由商品配送中心和连锁超市组成，其中商品配送中心包括采购、财务、配送等部门。为实现高效管理，设计了商品配送中心信息管理系统，其主要功能描述如下：

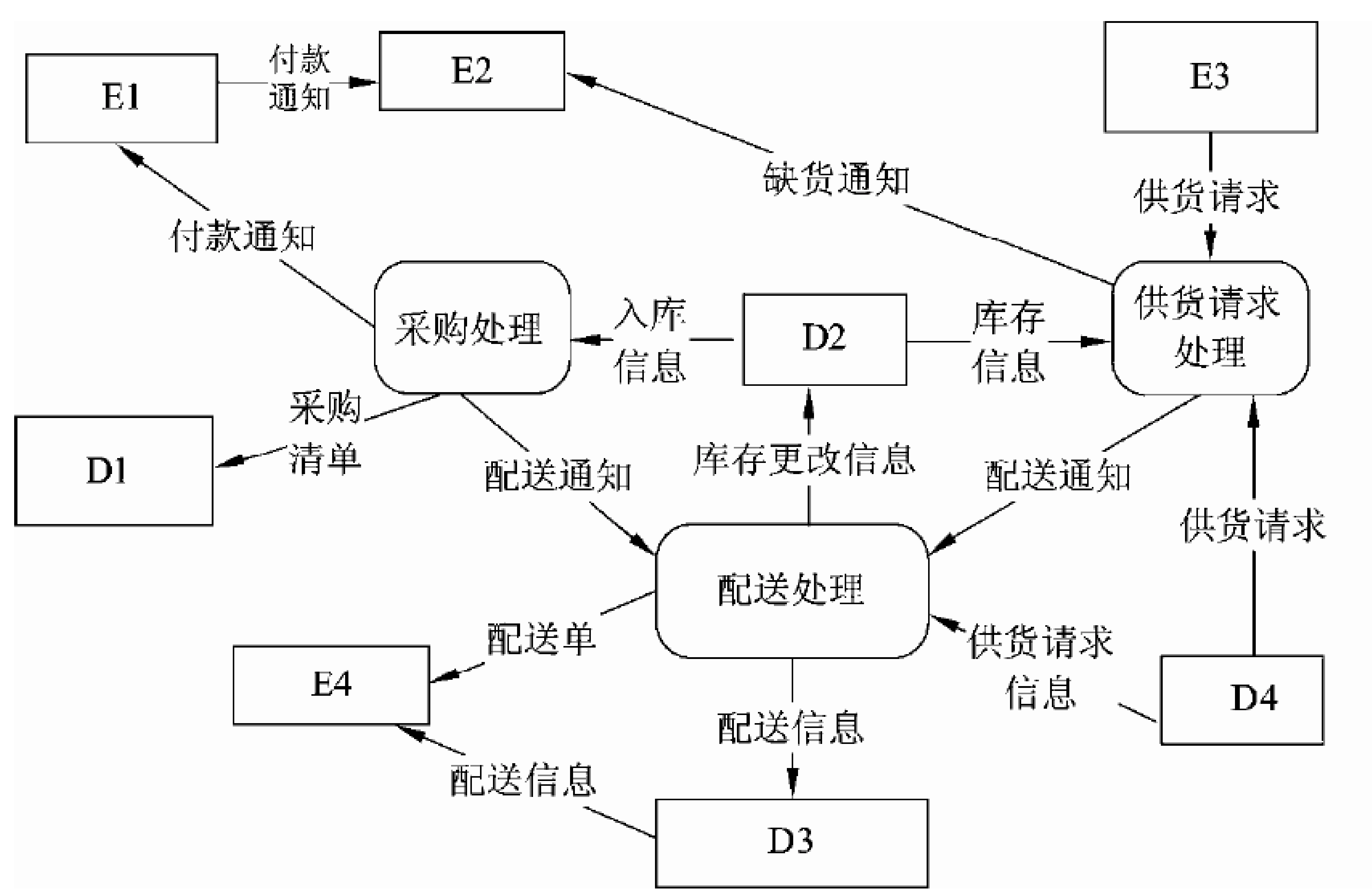
1．系统接收由连锁超市提出的供货请求，并将其记录到供货请求记录文件。

2．在接到供货请求后，从商品库存记录文件中进行商品库存信息查询。如果库存满足供货请求，则给配送处理发送配送通知；否则，向采购部门发出缺货通知。

3．配送处理接到配送通知后，查询供货请求记录文件，更新商品库存记录文件，并向配送部门发送配送单，在配送货品的同时记录配送信息至商品配送记录文件。

4．采购部门接到缺货通知后，与供货商洽谈，进行商品采购处理，合格商品入库，并记录采购清单至采购清单记录文件、向配送处理发出配送通知，同时通知财务部门给供货商支付货款。

该系统采用结构化方法进行开发，得到待修改的数据流图如下图所示。



数据流图

【问题 1】（8 分）

使用说明中的词语，给出上图中外部实体 E1 至 E4 的名称和数据存储 D1 至 D4 的名称。

【问题 2】（7 分）

以上数据流图中存在四处错误数据流，请指出各自的起点和终点；若将上述四条错误数据流删除，为保证数据流图的正确性，应补充三条数据流，请给出所补充数据流的起点和终点。（起点和终点请采用上述数据流图中的符号或名称）

错误数据流

起点	终点

补充的数据流

起点	终点

试题一分析

本题考查 DFD 的分析与设计。

【问题 1】

本问题考查 DFD 中的外部实体和数据存储，由于在题干中已经提到“系统接收由连锁超市提出的供货请求，并将其记录到供货请求记录文件”，因此可以明确出“连锁超市”外部实体和“供货请求记录文件”数据存储；对应到 DFD 图中为 E3 和 D4。描述中的第二项提出“从商品库存记录文件中进行商品库存信息查询。如果库存满足供货请求，则给配送处发送配送通知；否则，向采购部门发出缺货通知”，因为配送通知需要发送到采购部门，因此采购部门将成为系统的外部实体；同时，商品库存记录文件能够提供库存信息，所以 DFD 图中 E2 和 D2 分别为采购部门和商品配送记录文件。第三项需求“配送处理接到配送通知后，查询供货请求记录文件，更新商品库存记录文件，并向配送部门发送配送单，在配送货品的同时记录配送信息至商品配送记录文件”，所以配送处理需要查询供货请求记录文件，更新商品库存记录文件与商品配送记录文件，因此 D3 为商品配送记录文件；采购处理需要记录采购清单同时通知财务部门，所以 E1 应该为财务部门，D1 为采购清单记录文件，剩下的 E4 则为配送部门。

DFD 中出现的错误数据流为：E1 到 E2，E1 与 E2 的数据流不属于系统的范围；D3

到 E4，多余的数据流；D2 到采购处理，数据流方向错误；D4 到供货请求处理，数据流方向错误。

需要补充的数据流为：E2 到采购处理，因为 E2 是采购部门，采购部门需要给采购处提供入库商品信息；采购处到 D2 需要一条数据流，因为采购处理需要更改库存信息；供货请求处理到 D4 需要一条数据流，因为供货请求处理需要记录供货请求信息。

参考答案

【问题 1】

E1：财务部门 E2：采购部门
E3：连锁超市 E4：配送部门
D1：采购清单记录文件 D2：商品库存记录文件
D3：商品配送记录文件 D4：供货请求记录文件

【问题 2】

错误数据流

起点	终点
E1	E2
D3	E4
D2	采购处理
D4	供货请求处理

补充的数据流

起点	终点
E2	采购处理
采购处理	D2
供货请求处理	D4

试题二（共 15 分）

阅读下列说明，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某集团公司拥有多个大型连锁商场，公司需要构建一个数据库系统以方便管理其业务运作活动。

【需求分析结果】

1．商场需要记录的信息包括商场编号（编号唯一），商场名称，地址和联系电话。某商场信息如下表所示。

商场信息表

商场编号	商场名称	地址	联系电话
PS2101	淮海商场	淮海中路 918 号	021-64158818
PS2902	西大街商场	西大街时代盛典大厦	029-87283220
PS2903	东大街商场	碑林区东大街 239 号	029-87450287
PS2901	长安商场	雁塔区长安中路 38 号	029-85264953

2．每个商场包含有不同的部门，部门需要记录的信息包括部门编号（集团公司分配），部门名称，位置分布和联系电话。某商场的部门信息如下表所示。

部门信息表

部门编号	部门名称	位置分布	联系电话
DT002	财务部	商场大楼六层	82504342
DT007	后勤部	商场地下副一层	82504347
DT021	安保部	商场地下副一层	82504358
DT005	人事部	商场大楼六层	82504446
DT001	管理部	商场裙楼三层	82504668

3．每个部门雇用多名员工处理日常事务，每名员工只能隶属于一个部门（新进员工在培训期不隶属于任何部门）。员工需要记录的信息包括员工编号（集团公司分配），姓名，岗位，电话号码和工资。员工信息如下表所示。

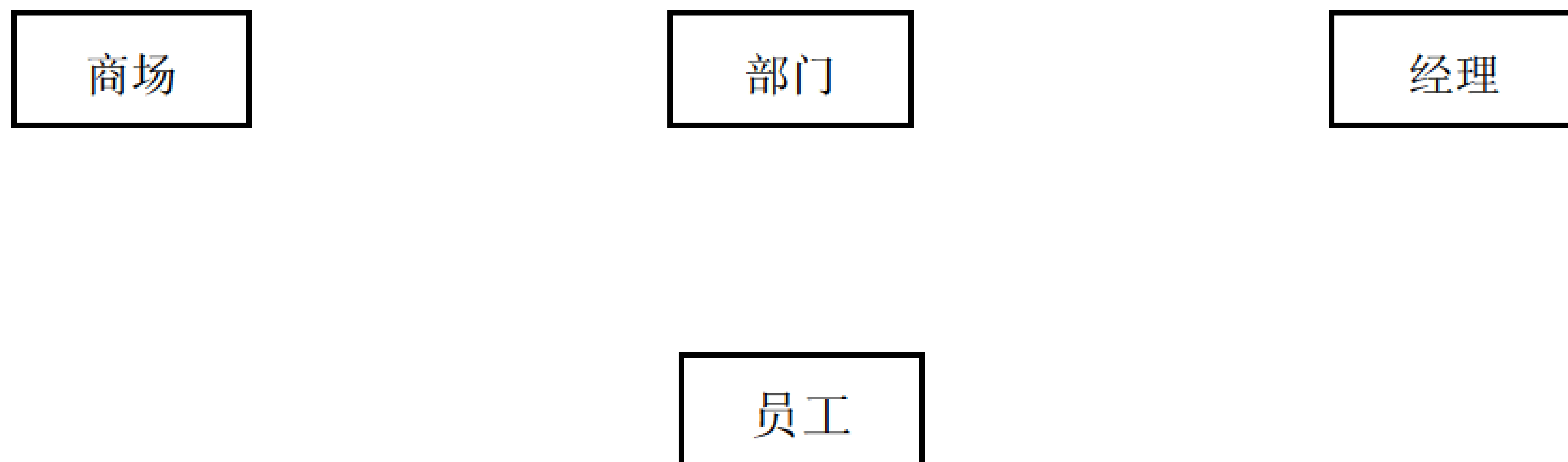
员工信息表

员工编号	姓名	岗位	电话号码	工资
XA3310	周　超	理货员	13609257638	1500.00
SH1075	刘　飞	防损员	13477293487	1500.00
XA0048	江雪花	广播员	15234567893	1428.00
BJ3123	张正华	部门主管	13345698432	1876.00

4．每个部门的员工中有一名是经理，每个经理只能管理一个部门，系统需要记录每个经理的任职时间。

【概念模型设计】

根据需求阶段收集的信息，设计的实体联系图和关系模式（不完整）如下：



实体联系图

【关系模式设计】

商场（商场编号，商场名称，地址，联系电话）
部门（部门编号，部门名称，位置分布，联系电话，____（a）____）
员工（员工编号，员工姓名，岗位，电话号码，工资，____（b）____）
经理（ （c） ，任职时间）

【问题 1】（6 分）

根据问题描述，补充四个联系，完善图 2-1 的实体联系图。联系名可用联系 1、联系 2、联系 3 和联系 4 代替，联系的类型分为 1:1、1:n 和 m:n。

【问题 2】（6 分）

根据实体联系图，将关系模式中的空（a）～（c）补充完整，并分别给出部门、员工和经理关系模式的主键和外键。

【问题 3】（3 分）

为了使商场有紧急事务时能联系到轮休的员工，要求每位员工必须且只能登记一位紧急联系人的姓名和联系电话，不同的员工可以登记相同的紧急联系人。则在图 2-1 中还需添加的实体是___（1）___，该实体和图 2-1 中的员工存在___（2）___联系（填写联系类型）。给出该实体的关系模式。

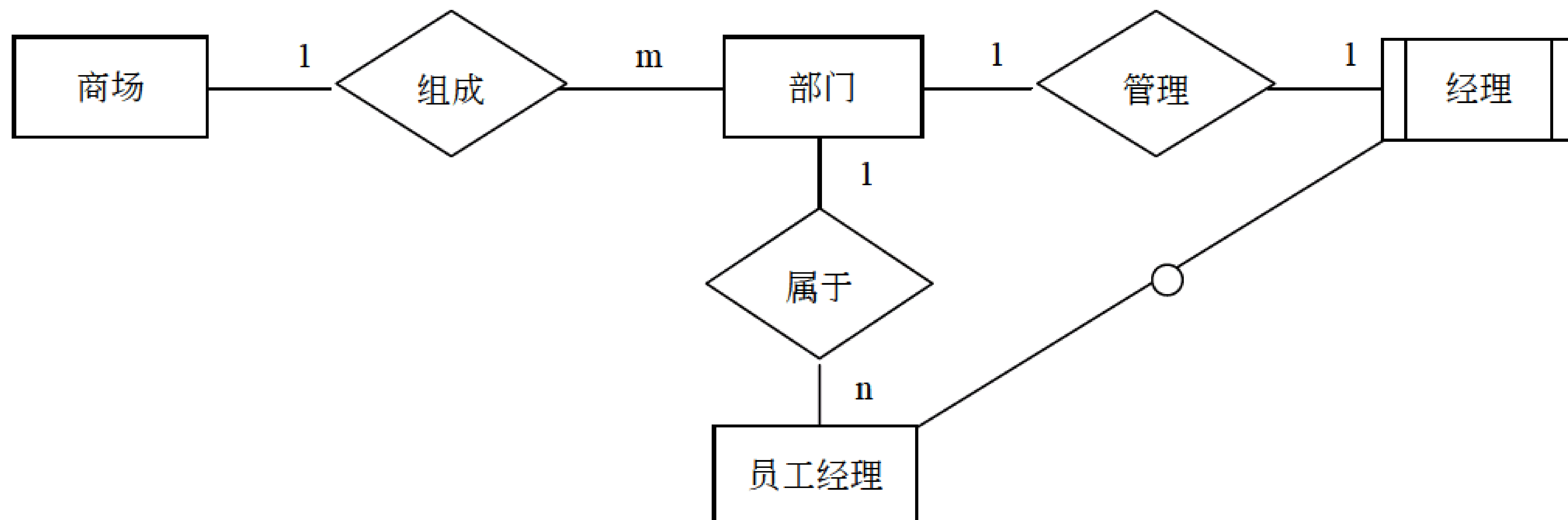
试题二分析

本题考查数据库概念结构设计及概念结构向逻辑结构转换的过程。

此类题目要求考生认真阅读题目对现实问题的描述，经过分类、聚集和概括等方法从中确定实体及其联系。题目已经给出了 4 个实体，需要根据需求描述给出实体间的联系。

【问题 1】

由“每个商场包含有不同的部门”可知商场与部门间为 1:m 联系；由“每个部门雇用了多名员工处理日常事务”可知部门与员工间为 1:n 联系；由“每个部门的员工中有一个经理……每个经理只能管理一个部门”可知部门与经理间为 1:1 联系，并且员工是经理的超类型，经理是员工的子类型。



【问题 2】

商场的属性信息中，商场编号由集团公司分配，不会重复，可作为商场的主键属性；部门的属性信息中，部门编号由集团公司分配，不会重复，可作为部门的主键属性，商场与部门的联系需要通过将商场的主键（商场编号）加入到部门中来表达；员工的属性信息中，员工编号由集团公司分配，不会重复，可作为员工的主键属性，部门与员工的联系需要通过将部门的主键（部门编号）加入到员工中来表达；经理除了包含员工的属性信息外，还需要任职时间属性。完整的关系模式如下：

商场（商场编号，商场名称，地址，联系电话）
部门（部门编号，部门名称，位置分布，联系电话，商场编号）
员工（员工编号，姓名，岗位，电话号码，工资，部门编号）
经理（员工编号，任职时间）

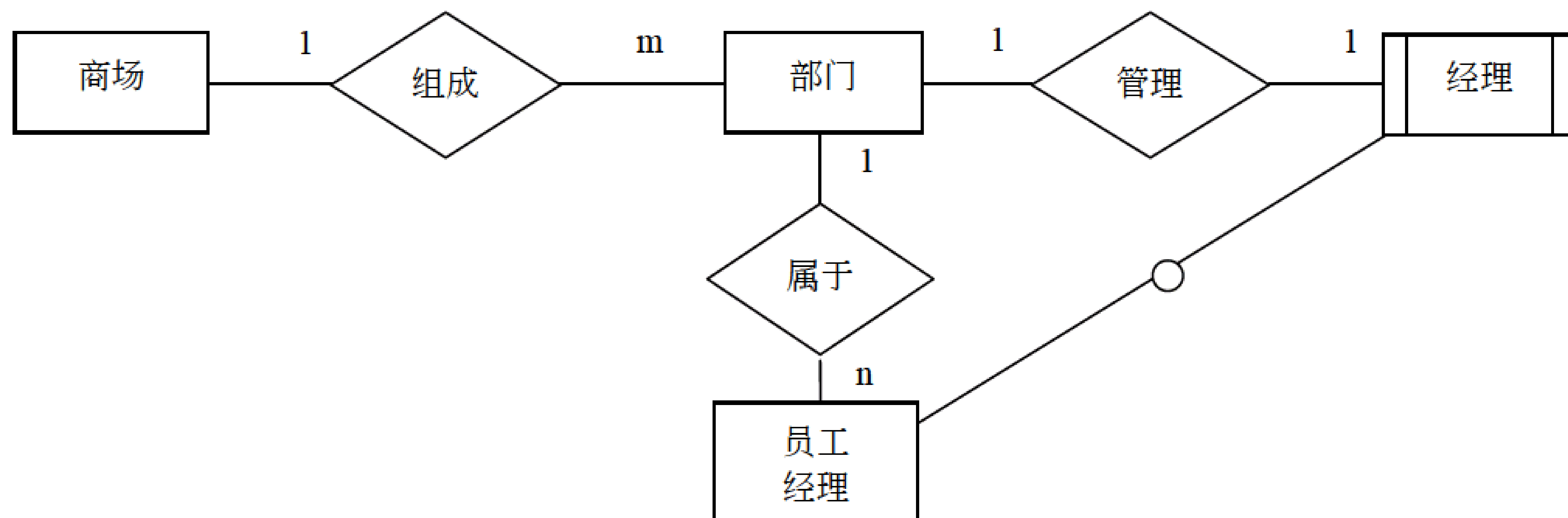
【问题 3】

员工的紧急联系人信息通过添加紧急联系人关系来实现，由“每位员工必须且只能登记一位紧急联系人的姓名和联系电话”，但可能存在多位员工登记同一位家属，可知员工与家属间为 n:1 联系；由“不同员工可以登记相同的紧急联系人”可知，员工编号可作为家属的主键属性。所以需要添加的关系模式如下：

紧急联系人（员工编号，姓名，联系电话）

参考答案

【问题 1】（图中的 m、n 也可用*表示，对联系名称可不做要求，但不能出现重名）



【问题 2】

（a）商场编号
（b）部门编号
（c）员工编号
部门关系模式的主键：部门编号
　　　　　　　外键：商场编号
员工关系模式的主键：员工编号

外键：部门编号

经理关系模式的主键：员工编号

外键：员工编号

【问题 3】

（d）紧急联系人　　　　（e）1: n

关系模式：紧急联系人（员工编号，姓名，联系电话）

试题三（共 15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某银行计划开发一个自动存提款机模拟系统（ATM System）。系统通过读卡器（CardReader）读取 ATM 卡；系统与客户（Customer）的交互由客户控制台（Customer-Console）实现；银行操作员（Operator）可控制系统的启动（System Startup）和停止（System Shutdown）；系统通过网络和银行系统（Bank）实现通信。

当读卡器判断用户已将 ATM 卡插入后，创建会话（Session）。会话开始后，读卡器进行读卡，并要求客户输入个人验证码（PIN）。系统将卡号和个人验证码信息送到银行系统进行验证。验证通过后，客户可从菜单选择如下事务（Transaction）：

1．从 ATM 卡账户取款（Withdraw）；

2．向 ATM 卡账户存款（Deposit）；

3．进行转账（Transfer）；

4．查询（Inquire）ATM 卡账户信息。

一次会话可以包含多个事务，每个事务处理也会将卡号和个人验证码信息送到银行系统进行验证。若个人验证码错误，则转个人验证码错误处理（Invalid PIN Process）。每个事务完成后，客户可选择继续上述事务或退卡。选择退卡时，系统弹出 ATM 卡，会话结束。

系统采用面向对象方法开发，使用 UML 进行建模。系统的顶层用例图如图 3-1 所示，一次会话的序列图（不考虑验证）如图 3-2 所示。

【问题 1】（7 分）

根据说明中的描述，给出图 3-1 中 A1 和 A2 所对应的参与者，U1 至 U3 所对应的用例，以及该图中空__（1）__所对应的关系。（U1 至 U3 的可选用例包括：Session、Transaction、Insert Card、Invalid PIN Process 和 Transfer）

【问题 2】（6 分）

根据说明中的描述，使用消息名称列表中的英文名称，给出图 3-2 中 6～9 对应的消息。

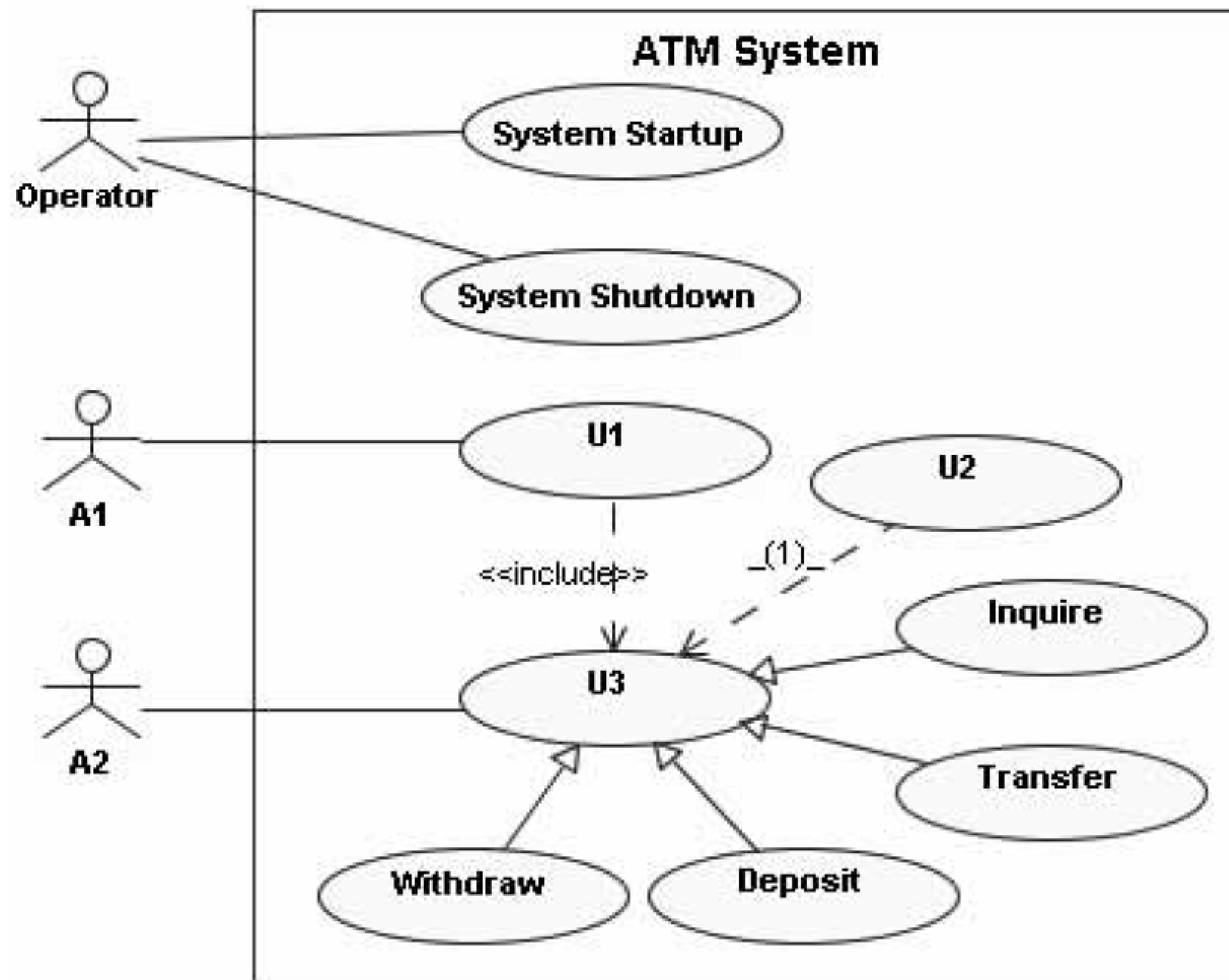


图 3-1　ATM 系统顶层用例图

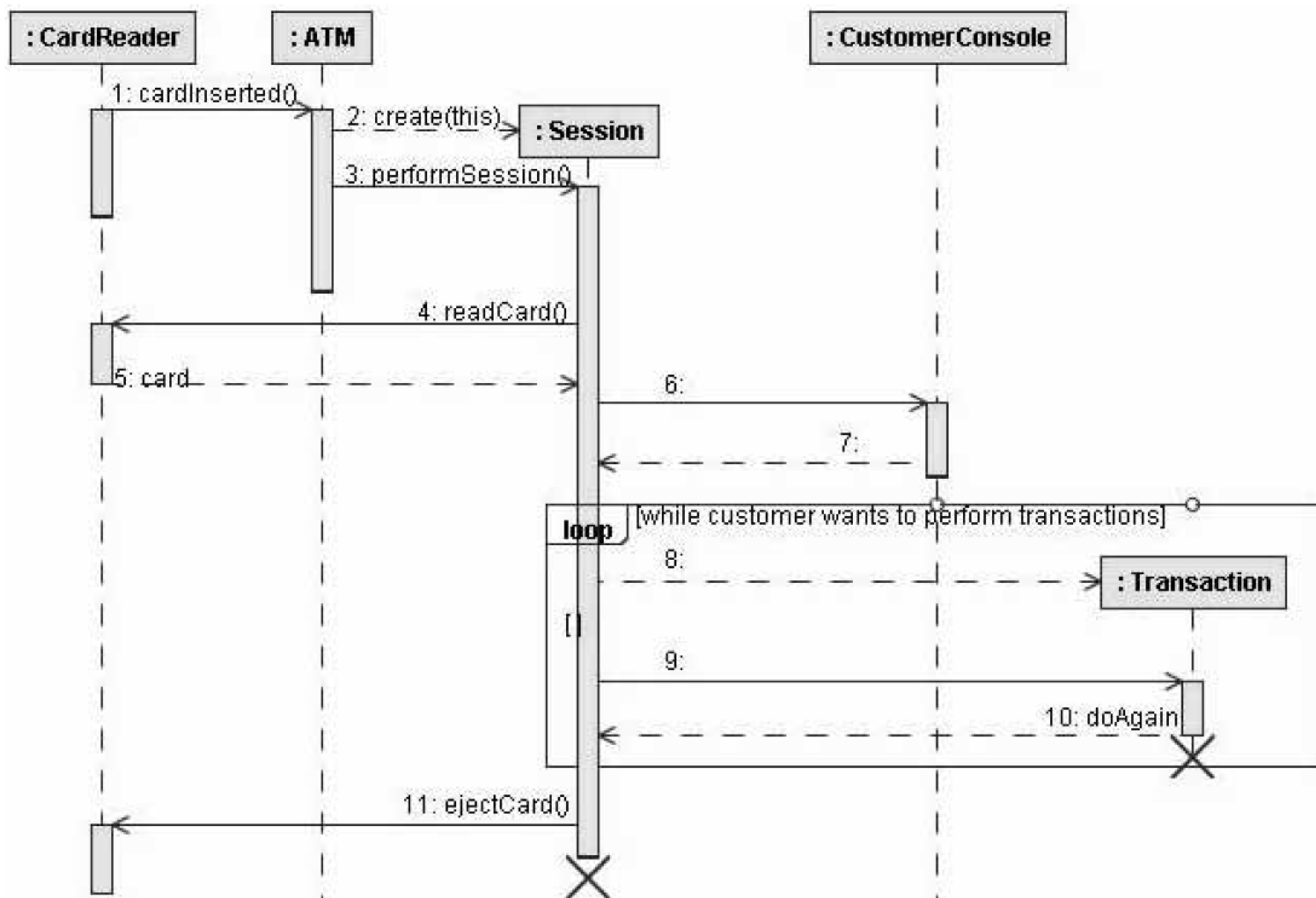


图 3-2　一次会话的序列图（无验证消息）

可能的消息名称列表

名　称	说　明	名　称	说　明
cardInserted()	ATM 卡已插入	performTransaction()	执行事务
performSession()	执行会话	readCard()	读卡
readPIN()	读取个人验证码	PIN	个人验证码信息
creat(atm, this, card, pin)	为当前会话创建事务	create(this)	为当前 ATM 创建会话
card	ATM 卡信息	doAgain	执行下一个事务
ejectCard()	弹出 ATM 卡		

【问题 3】（2 分）

解释图 3-1 中用例 U3 和用例 Withdraw、Deposit 等四个用例之间的关系及其内涵。

试题三分析

本题涉及面向对象系统开发时的 UML 用例图、序列图以及用例之间的关系。

【问题 1】

构建用例图时，常用的方式是先识别参与者，然后确定用例以及用例之间的关系。

识别参与者时，考查和系统交互的人员和外部系统。本题中，与系统交互的人员包括客户（Customer）和银行操作员（Operator），与本模拟系统交互的外部系统包括银行系统（Bank）。

考查用例时，通过判断哪一个特定参与者发起或者触发了与系统的哪些交互，来识别用例并建立和参与者之间的关联。考查用例之间的关系时，<<include>>（包含）定义了用例之间的包含关系，用于一个用例包含另一个用例的行为的建模；如果可以从一个用例的执行中，在需要时转向执行另一个用例，执行完返回之前的用例继续执行，用例间即存在<<extend>>关系。

本题中，客户一旦插卡成功，系统就创建会话（Session），会话中可以执行用户从菜单选择的 Withdraw、Deposit、Transfer 和 Inquire 等事务（Transaction）。由图中 U3 和 Withdraw 之间的扩展关系，可知 U3 为 Transaction；又由 U1 和 U3 之间的<<include>>关系，得知 U1 为 Session，进而判定图中 A1 为 Customer，A2 为 Bank。每个事务处理也会将卡号和个人验证码信息送到银行系统进行验证，若个人验证码错误，则转个人验证码错误处理（Invalid PIN Process，图中 U2），所以（1）处应填<<extend>>。

【问题 2】

序列图是场景的图形化表示，描述了以时间顺序组织的对象之间的交互活动。构造序列图时遵循如下指导原则：确定顺序图的范围，描述这个用例场景或一个步骤；绘制参与者和接口类，如果范围包括这些内容的话；沿左手边列出用例步骤；对控制器类及必须在顺序中协作的每个实体类，基于它拥有的属性或已经分配给它的行为绘制框；为

持续类和系统类绘制框；绘制所需消息，并把每条消息指到将实现响应消息的责任的类上；添加活动条指示每个对象实例的生命期；为清晰起见，添加所需的返回消息；如果需要，为循环、可选步骤和替代步骤等添加框架。

本题中，根据说明中的描述，从 ATM 机判断卡已插入（cardInserted()）开始会话，即为当前 ATM 创建会话（create(this)）并开始执行会话（performSession()）；读卡器读卡（readCard()）获得ATM卡信息（card），然后从控制台读取个人验证码输入（readPIN()，图中标号 6 处）并获得个人验证码信息（PIN，图中标号 7 处）；然后根据用户选择启动并执行事务，即为当前会话创建事务（creat(atm, this, card, pin)，图中标号 8 处）和执行事务（performTransaction()，图中标号 9 处）；可以选择继续执行某个事务（doAgain）循环，或者选择退卡（ejectCard()）。

【问题 3】

用例之间的继承关系表示子类型“是一种”父类型。其中父类型通常是一个抽象泛化用例，具有子类型共有的属性和行为，每个具体的子类型继承它，并实现适合自己的特定的操作。

本题中 Transaction 和 Withdraw、Deposit 等四个用例之间的关系即为继承关系，Transaction 即是一个抽象泛化用例，具有其他事务类型共有的属性和行为，每个具体的事务类型继承它，并实现适合自己的特定的操作。

参考答案

【问题 1】

A1：Customer　　A2：Bank　　U1：Session

U2：Invalid PIN Process　　U3：Transaction　　（1）：<<extend>>

【问题 2】

6：readPIN()　　7：PIN　　8：creat(atm, this, card, pin)

9：performTransaction()

【问题 3】

Transaction 是一个抽象泛化用例，具有其他事务类型共有的属性和行为，每个具体的事务类型继承它，并实现适合自己的特定的操作。

试题四（共 15 分）

阅读下列说明，回答问题 1 和问题 2，将解答填入答题纸的对应栏内。

【说明】

现需在某城市中选择一个社区建一个大型超市，使该城市的其他社区到该超市的距离总和最小。用图模型表示该城市的地图，其中顶点表示社区，边表示社区间的路线，边上的权重表示该路线的长度。

现设计一个算法来找到该大型超市的最佳位置：即在给定图中选择一个顶点，使该顶点到其他各顶点的最短路径之和最小。算法首先需要求出每个顶点到其他任一顶点的

最短路径，即需要计算任意两个顶点之间的最短路径；然后对每个顶点，计算其他各顶点到该顶点的最短路径之和；最后，选择最短路径之和最小的顶点作为建大型超市的最佳位置。

【问题 1】（12 分）

本题采用 Floyd-Warshall 算法求解任意两个顶点之间的最短路径。已知图 G 的顶点集合为 $V=\{1,2,\cdots,n\}$，$W=\{w_{ij}\}_{n*n}$ 为权重矩阵。设 $d_{ij}^{(k)}$ 为从顶点 i 到顶点 j 的一条最短路径的权重。当 $k=0$ 时，不存在中间顶点，因此 $d_{ij}^{(0)}=w_{ij}$；当 $k>0$ 时，该最短路径上所有的中间顶点均属于集合 $\{1,2,\cdots,k\}$。若中间顶点包括顶点 k，则 $d_{ij}^{(k)}=d_{ik}^{(k-1)}+d_{kj}^{(k-1)}$；若中间顶点不包括顶点 k，则 $d_{ij}^{(k)}=d_{ij}^{(k-1)}$。于是得到如下递归式。

$$d_{ij}^{(k)}=\begin{cases} w_{ij} & k=0 \\ \min(d_{ij}^{(k-1)},d_{ik}^{(k-1)}+d_{kj}^{(k-1)}) & k>0 \end{cases}$$

因为对于任意路径，所有的中间顶点都在集合 $\{1,2,\cdots,n\}$ 内，因此矩阵 $D^{(n)}=\{d_{ij}^{(n)}\}_{n*n}$ 给出了任意两个顶点之间的最短路径，即对所有 $i,j\in V$，$d_{ij}^{(n)}$ 表示顶点 i 到顶点 j 的最短路径。

下面是求解该问题的伪代码，请填充其中空缺的（1）至（6）处。伪代码中的主要变量说明如下：

W：权重矩阵

n：图的顶点个数

SP：最短路径权重之和数组，SP[i]表示顶点 i 到其他各顶点的最短路径权重之和，i 从 1 到 n

min_SP：最小的最短路径权重之和

min_v：具有最小的最短路径权重之和的顶点

i：循环控制变量

j：循环控制变量

k：循环控制变量

```
LOCATE  -SHOPPINGMALL(W, n)
1    D(0) = W
2    for ___(1)___
3        for i = 1 to n
4            for j = 1 to n
5            if d_ij^(k-1) ≤ d_ik^(k-1) + d_kj^(k-1)
6                    ___(2)___
7                else
8                    ___(3)___
```

```
9    for i = 1 to n
10       SP[i] = 0
11       for j = 1 to n
12           ___(4)___
13   min_SP = SP[1]
14   ___(5)___
15   for i = 2 to n
16       if min_SP > SP[i]
17           min_SP = SP[i]
18           min_v = i
19   return ___(6)___
```

【问题 2】（3 分）

问题 1 中伪代码的时间复杂度为___(7)___（用 O 符号表示）。

试题四分析

本题考查的是算法的设计和分析技术。

【问题 1】

本问题考查算法流程。第（1）空表示主循环，k 是循环控制变量，故第（1）空填 k = 1 to n。第（2）和（3）空根据题意和递归式，可分别得到答案为 $d_{ij}^{(k)} = d_{ij}^{(k-1)}$ 和 $d_{ij}^{(k)} = d_{ik}^{(k-1)} + d_{kj}^{(k-1)}$。计算了任意两个顶点之间的最短路径之后，对每个顶点，开始统计其到所有其他顶点的最短路径之和，因此第（4）空填 $SP[i] = SP[i] + d_{ij}^{(n)}$。第 13 和第 14 行初始化，假设最小的到所有其他顶点的最短路径之和为第一个顶点的最小路径之和，大型超市的最佳位置为第一个顶点，故第（5）空填 min_v = 1。最后要求返回大型超市的最佳位置，即到所有其他顶点的最短路径之和最小的顶点，故第（6）空填 min_v。

【问题 2】

本问题考查问题 1 中的伪代码第 2～8 行，计算任意两点之间的最短路径，有三重循环，故时间复杂度为 $O(n^3)$。第 9～12 行，计算每个点到任意其他点的最短路径之和，有两重循环，故时间复杂度为 $O(n^2)$。第 15～18 行，在所有点的最短路径之和中找到最小的最短路径之和，时间复杂度为 $O(n)$。故算法总的时间复杂度为 $O(n^3)$。

参考答案

【问题 1】

（1）k = 1 to n　　（2）$d_{ij}^{(k)} = d_{ij}^{(k-1)}$　　（3）$d_{ij}^{(k)} = d_{ik}^{(k-1)} + d_{kj}^{(k-1)}$

（4）$SP[i] = SP[i] + d_{ij}^{(n)}$　　（5）min_v = 1　　（6）min_v

【问题 2】

（7）$O(n^3)$

试题五（共 15 分）

阅读下列说明和 C 函数代码，将应填入 (n) 处的字句写在答题纸的对应栏内。

【说明】

对二叉树进行遍历是二叉树的一个基本运算。遍历是指按某种策略访问二叉树的每个节点，且每个节点仅访问一次的过程。函数 InOrder()借助栈实现二叉树的非递归中序遍历运算。

设二叉树采用二叉链表存储，节点类型定义如下：

```
typedef struct BtNode{
    ElemType  data;                /*节点的数据域，ElemType 的具体定义省略*/
    struct BtNode *lchild,*rchild;   /*节点的左、右孩子指针域*/
}BtNode, *BTree;
```

在函数 InOrder()中，用栈暂存二叉树中各个节点的指针，并将栈表示为不含头节点的单向链表（简称链栈），其节点类型定义如下：

```
typedef struct StNode{          /*链栈的节点类型*/
      BTree elem;               /*栈中的元素是指向二叉链表节点的指针*/
      struct StNode *link;
}StNode;
```

假设从栈顶到栈底的元素为 e_n、e_{n-1}、…、e_1，则不含头节点的链栈示意图如图 5-1 所示。

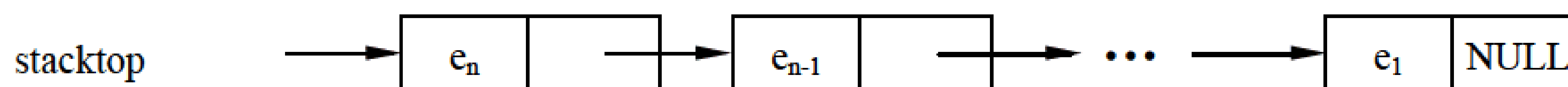


图 5-1　链栈示意图

【C 函数】

```
int  InOrder(BTree root)          /*实现二叉树的非递归中序遍历*/
{
    BTree ptr;                    /*ptr 用于指向二叉树中的节点*/
    StNode *q;                    /*q 暂存链栈中新创建或待删除的节点指针*/
    StNode *stacktop = NULL;      /*初始化空栈的栈顶指针 stacktop*/
    ptr = root;                   /*ptr 指向二叉树的根节点*/
    while ( ___(1)___ || stacktop != NULL) {
    while (ptr != NULL) {
        q = (StNode *)malloc(sizeof(StNode));
        if (q == NULL)
return -1;
```

```
            q->elem = ptr;
            ___(2)___;
             stacktop = q;          /*stacktop 指向新的栈顶*/
             ptr = ___(3)___;       /*进入左子树*/
        }
             q = stacktop;
             ___(4)___;             /*栈顶元素出栈*/
             visit(q);              /*visit 是访问节点的函数，其具体定义省略*/
             ptr = ___(5)___;       /*进入右子树*/
             free(q);               /*释放原栈顶元素的节点空间*/
        }
        return 0;
    }/*InOrder*/
```

试题五分析

本题考查基本数据结构和 C 语言程序设计能力。

对非空二叉树进行中序遍历的方法是：先中序遍历根节点的左子树，然后访问根节点，最后中序遍历根节点的右子树。用递归方式描述的算法如下：

```
void In_Order_Traversing(BiTree root)
{   //root 是指向二叉树根节点的指针
    if (root != NULL) {
        In_Order_Traversing(root->LeftChild);
        visit(root);
        In_Order_Traversing(root->RightChild);
    }
}
```

从以上算法的执行过程可知，从树根出发进行遍历时，递归调用 In_Order_Traversing(root->LeftChild)使得遍历过程沿着左孩子分支一直走向下层节点，直到到达二叉树中最左下方的节点（设为 f）的空左子树为止，然后返回 f 节点，再由递归调用 In_Order_Traversing(root->RightChild)进入 f 的右子树，并重复以上过程。在递归算法执行过程中，辅助实现递归调用和返回处理的控制栈实际上起着保存从根节点到当前节点的路径信息。

用非递归算法实现二叉树的中序遍历时，可以由一个循环语句实现从指定的根节点出发，沿着左孩子分支一直到头（到达一个没有左子树的节点）的处理，从根节点到当前节点的路径信息（节点序列）可以明确构造一个栈来保存。

本题目的难点在于将栈的实现和使用混合在一起来处理，而且栈采用单链表存储结构。下面分析题中给出的代码。

空（1）是遍历的条件之一，由于另外一个条件 stacktop != NULL 初始时是不成立的，因此空（1）所表示的条件必须满足，由于是对非空二叉树进行遍历，显然该条件代表二叉树非空，即 ptr != NULL 或其等价表示形式。

临时指针 ptr 初始时指向整个二叉树的根节点，此后用以下代码表示一直沿左孩子指针链向下走的处理，临时指针 q 用于在链栈中加入新元素时使用。处理思路是：若当前节点有左子树，则将当前节点的指针存入栈中，然后进入当前节点的左子树。入栈时，先申请元素在链栈中的节点空间，然后设置节点数据域的值（即当前节点的指针），最后将新申请的节点加入链栈首部。

```
while (ptr != NULL) {
        q = (StNode *)malloc(sizeof(StNode)); /*为新入栈的元素创建节点*/
        if (q == NULL)             /*若创建新节点失败，则退出*/
            return -1;
        q->elem = ptr;             /*在栈顶保存指向当前节点的指针*/
        q->link = stacktop ;       /*新节点加入栈顶*/
stacktop = q;                      /*更新栈顶指针，即 stacktop 指向新的栈顶*/
ptr = ptr->lchild;                 /*进入当前节点的左子树*/
}
```

当上述过程进入一棵空的子树时（ptr 为空指针），循环结束。此后，应该从空的子树返回其父节点并进行访问。由于进入空的左子树前已将其父节点指针压入栈中，因此，栈顶元素即为该父节点，对应的处理就是弹栈。相应地，在链栈中要删除表头节点并释放节点空间：

```
q = stacktop;                 /*q 指向链栈中需要删除的节点，即栈顶元素*/
 stack = stacktop->link ;     /*栈顶元素出栈*/
visit(q);                     /*访问节点*/
free(q);                      /*释放节点空间*/
```

由于还需要通过 q 指针进入被删除节点的右子树，因此，释放节点空间的操作 free(q) 操作之前，使 ptr 指向 q 所指节点的右子树指针，以得到被删除节点的数据域信息，即空（5）所在语句 ptr = q->elem->rchild。

指针是 C 语言中灵活且非常强大的工具，是否熟练掌握 C 语言的判断条件之一就是对指针的理解和使用。软件设计师需要熟练掌握这些内容。

参考答案

（1）ptr != NULL，或 ptr !=0，或 ptr

（2）q->link = stacktop

（3）ptr->lchild

（4）stacktop = stacktop->link，或 stacktop = q->link

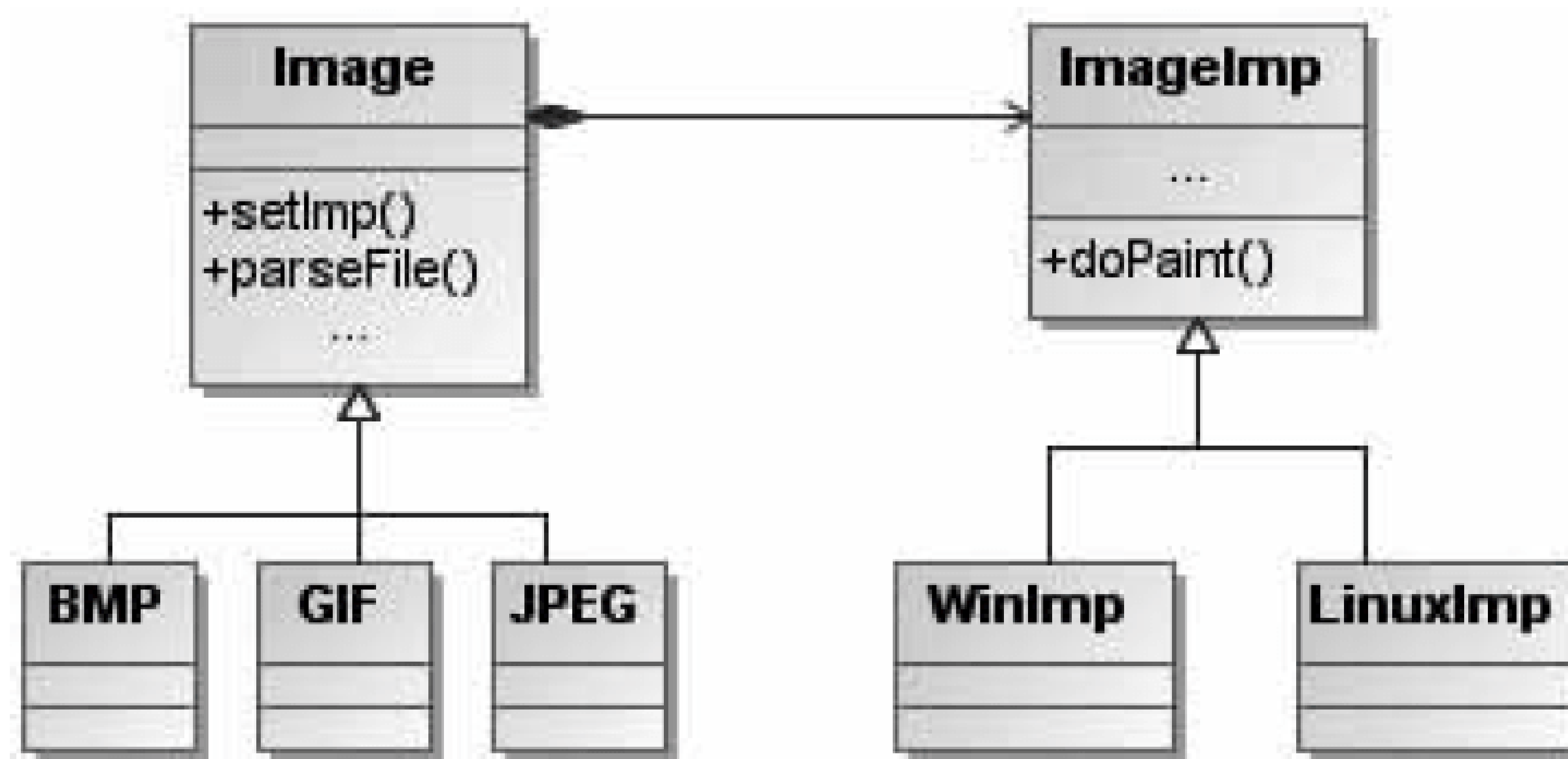
（5）q->elem->rchild

试题六（共 15 分）

阅读下列说明和 C++代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

现欲实现一个图像浏览系统，要求该系统能够显示 BMP、JPEG 和 GIF 三种格式的文件，并且能够在 Windows 和 Linux 两种操作系统上运行。系统首先将 BMP、JPEG 和 GIF 三种格式的文件解析为像素矩阵，然后将像素矩阵显示在屏幕上。系统需具有较好的扩展性以支持新的文件格式和操作系统。为满足上述需求并减少所需生成的子类数目，采用桥接（Bridge）设计模式进行设计，所得类图如下图所示。



类图

采用该设计模式的原因在于：系统解析 BMP、GIF 与 JPEG 文件的代码仅与文件格式相关，而在屏幕上显示像素矩阵的代码则仅与操作系统相关。

【C++代码】

```
class Matrix{    //各种格式的文件最终都被转化为像素矩阵
    //此处代码省略
};
class ImageImp{
public:
    virtual void doPaint(Matrix m) = 0;  //显示像素矩阵 m
};

class WinImp : public ImageImp{
public:
    void doPaint(Matrix m){ /*调用 Windows 系统的绘制函数绘制像素矩阵*/ }
};
```

```
class LinuxImp : public ImageImp{
public:
    void doPaint(Matrix m){ /*调用 Linux 系统的绘制函数绘制像素矩阵*/ }
};

class Image {
public:
    void setImp(ImageImp *imp){___(1)___ = imp;}
    virtual void parseFile(string fileName) = 0;
protected:
    ___(2)___ *imp;
};

class  BMP : public Image{
public:
    void parseFile(string fileName){
        //此处解析 BMP 文件并获得一个像素矩阵对象 m
        ___(3)___;// 显示像素矩阵 m

    }
};
class  GIF : public Image{
    //此处代码省略
};

class  JPEG : public Image{
    //此处代码省略
};

void main(){
    //在 Windows 操作系统上查看 demo.bmp 图像文件
    Image *image1 = ___(4)___;
    ImageImp *imageImp1 = ___(5)___;
    ___(6)___;
```

```
        image1->parseFile("demo.bmp");

    }
```

现假设该系统需要支持 10 种格式的图像文件和 5 种操作系统，不考虑类 Matrix，若采用桥接设计模式则至少需要设计 __(7)__ 个类。

试题六分析

根据题目描述，在设计该图像显示系统时主要分为两个步骤：一是读取各种文件并将文件内容转换成像素矩阵，因为各种图片格式不同，因此需要针对每一种图片格式编写文件读取代码，而该代码与操作系统平台无关。将像素矩阵显示到屏幕上时，由于和操作系统相关，因此需要把该代码和读取文件代码相分离。设计中的 Image 类表示抽象的图像概念，Image 类中就包含了读取文件接口和设置实现平台接口；Image 的子类 BMP、GIF 和 JPEG 分别负责读取各种不同格式的文件；ImageImp 的主要任务是将像素矩阵显示在屏幕上，因此，它存在两个子类，分别实现 Windows 系统和 Linux 系统上的图像显示代码。空缺（1）处主要是设置将在哪个平台上进行实现，因此该处应该存储参数所传递的对象，由于该类的成员变量也是 imp，与参数相同，因此需要填写 this->imp；同理，该成员变量的类型和参数的类型应该保持相同，空（2）处应该填写 ImageImp；空（3）处需要根据 imp 成员变量存储的实现对象来显示图像；在空（4）处需要生成一个 BMP 对象；由于需要在 Windows 平台上实现，因此空（5）处需要生成一个 WinImp 对象，同时，还需设置该 BMP 对象，应采用 WinImp 对象来实现显示。采用桥接模式能够将文件分析代码和图像显示代码分解在不同的类层次结构中，如果不考虑中间使用的 Matrix 等类，那么最后需要设计的类包括两个父类，对应文件格式子类，对应操作系统平台类，因此 10 种图像格式和 5 种操作系统需要 17 个类。

参考答案

（1）this->imp　（2）ImageImp　（3）imp->doPaint(m)　（4）new BMP()

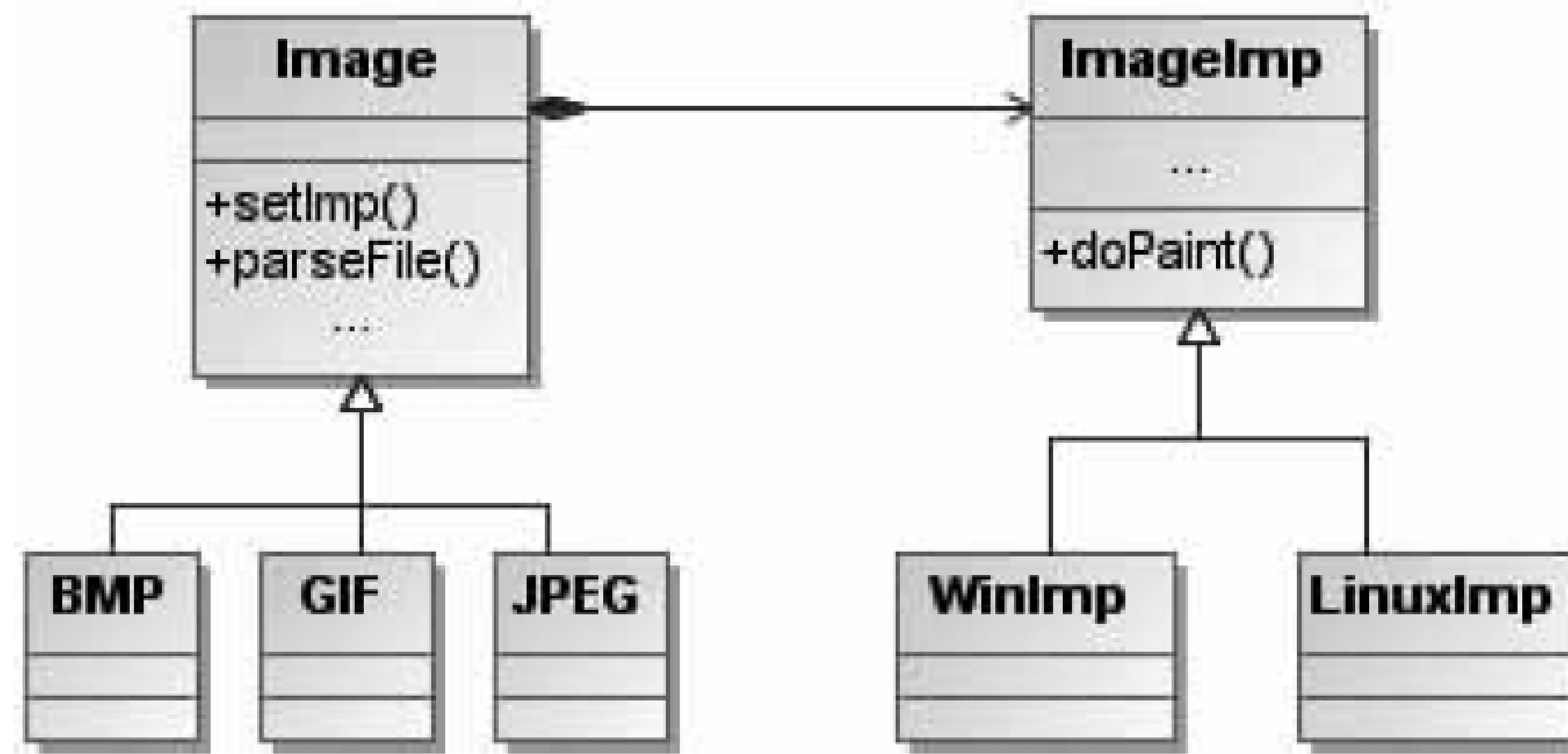
（5）new WinImp()　（6）image1->setImp(imageImp1)　（7）17

试题七（共 15 分）

阅读下列说明和 Java 代码，将应填入 __(n)__ 处的字句写在答题纸的对应栏内。

【说明】

现欲实现一个图像浏览系统，要求该系统能够显示 BMP、JPEG 和 GIF 三种格式的文件，并且能够在 Windows 和 Linux 两种操作系统上运行。系统首先将 BMP、JPEG 和 GIF 三种格式的文件解析为像素矩阵，然后将像素矩阵显示在屏幕上。系统需具有较好的扩展性以支持新的文件格式和操作系统。为满足上述需求并减少所需生成的子类数目，采用桥接（Bridge）设计模式进行设计，所得类图如下图所示。



类图

采用该设计模式的原因在于：系统解析 BMP、GIF 与 JPEG 文件的代码仅与文件格式相关，而在屏幕上显示像素矩阵的代码则仅与操作系统相关。

【Java 代码】

```
class Matrix{    //各种格式的文件最终都被转化为像素矩阵
    //此处代码省略
};
abstract class ImageImp{
    public abstract void doPaint(Matrix m);  //显示像素矩阵 m
};

class WinImp extends ImageImp{
    public void doPaint(Matrix m){      /*调用 Windows 系统的绘制函数绘制像
素矩阵*/ }
};

class LinuxImp extends ImageImp{
    public void doPaint(Matrix m){/*调用 Linux 系统的绘制函数绘制像素矩阵*/}
};

abstract class Image {
    public void setImp(ImageImp imp){
        ___(1)___ = imp; }
    public abstract void parseFile(String fileName);
    protected ___(2)___ imp;
};

class  BMP extends Image{
    public void parseFile(String fileName){
            //此处解析 BMP 文件并获得一个像素矩阵对象 m
            ___(3)___;// 显示像素矩阵 m

        }
```

```
};
class  GIF extends Image{
//此处代码省略
};

class  JPEG extends Image{
     //此处代码省略
};

public class javaMain{
     public static void main(String[] args){
          //在 Windows 操作系统上查看 demo.bmp 图像文件
          Image image1 = ___(4)___;
          ImageImp imageImp1 = ___(5)___;
          ___(6)___;
          image1.parseFile("demo.bmp");
     }
}
```

现假设该系统需要支持 10 种格式的图像文件和 5 种操作系统，不考虑类 Matrix 和类 javaMain，若采用桥接设计模式则至少需要设计___(7)___个类。

试题七分析

根据题目描述，在设计该图像显示系统时主要分为两个步骤：一是读取各种文件并将文件内容转换为像素矩阵，因为各种图片格式不同，因此需要针对每一种图片格式编写文件读取代码，而该代码与操作系统平台无关。将像素矩阵显示到屏幕上时，由于和操作系统相关，因此需要把该代码和读取文件代码相分离。设计中的 Image 类表示抽象的图像概念，Image 类中就包含了读取文件接口和设置实现平台接口；Image 的子类 BMP、GIF 和 JPEG 分别负责读取各种不同格式的文件；ImageImp 的主要任务是将像素矩阵显示在屏幕上，因此，它存在两个子类，分别实现 Windows 系统和 Linux 系统上的图像显示代码。空缺（1）处主要是设置将在哪个平台上进行实现，因此该处应该存储参数所传递的对象，由于该类的成员变量也是 imp，与参数相同，因此需要填写 this.imp；同理，该成员变量的类型和参数的类型应该保持相同，空（2）处应该填写 ImageImp；空（3）处需要根据 imp 成员变量存储的实现对象来显示图像；在空（4）处需要生成一个 BMP 对象；由于需要在 Windows 平台上实现，因此空（5）处需要生成一个 WinImp 对象，同时，还需设置该 BMP 对象，应采用 WinImp 对象来实现显示。采用桥接模式能够将文件分析代码和图像显示代码分解在不同的类层次结构中，如果不考虑中间使用的 Matrix 等类，那么最后需要设计的类包括 2 个父类，对应文件格式数目的子类，对应操作系统数目的平台类，因此 10 种图像格式和 5 种操作系统需要 17 个类。

参考答案

（1）this.imp　（2）ImageImp　（3）imp.doPaint(m)　（4）new BMP()
（5）new WinImp()　（6）image1.setImp(imageImp1)　（7）17

第 3 章　2009 下半年软件设计师上午试题分析与解答

试题（1）

以下关于 CPU 的叙述中，错误的是__（1）__。

（1）A．CPU 产生每条指令的操作信号并将操作信号送往相应的部件进行控制

B．程序计数器 PC 除了存放指令地址，也可以临时存储算术/逻辑运算结果

C．CPU 中的控制器决定计算机运行过程的自动化

D．指令译码器是 CPU 控制器中的部件

试题（1）分析

本题考查计算机硬件组成基础知识。

CPU 是计算机的控制中心，主要由运算器、控制器、寄存器组和内部总线等部件组成。控制器由程序计数器、指令寄存器、指令译码器、时序产生器和操作控制器组成，它是发布命令的“决策机构”，即完成协调和指挥整个计算机系统的操作。它的主要功能有：从内存中取出一条指令，并指出下一条指令在内存中的位置；对指令进行译码或测试，并产生相应的操作控制信号，以便启动规定的动作；指挥并控制 CPU、内存和输入输出设备之间数据的流动。

程序计数器（PC）是专用寄存器，具有寄存信息和计数两种功能，又称为指令计数器，在程序开始执行前，将程序的起始地址送入 PC，该地址在程序加载到内存时确定，因此 PC 的初始内容即是程序第一条指令的地址。执行指令时，CPU 将自动修改 PC 的内容，以便使其保持的总是将要执行的下一条指令的地址。由于大多数指令都是按顺序执行的，因此修改的过程通常只是简单地对 PC 加 1。当遇到转移指令时，后继指令的地址根据当前指令的地址加上一个向前或向后转移的位移量得到，或者根据转移指令给出的直接转移的地址得到。

参考答案

（1）B

试题（2）

以下关于 CISC（Complex Instruction Set Computer，复杂指令集计算机）和 RISC（Reduced Instruction Set Computer，精简指令集计算机）的叙述中，错误的是__（2）__。

（2）A．在 CISC 中，其复杂指令都采用硬布线逻辑来执行

B．采用 CISC 技术的 CPU，其芯片设计复杂度更高

C．在 RISC 中，更适合采用硬布线逻辑执行指令

D．采用 RISC 技术，指令系统中的指令种类和寻址方式更少

试题（2）分析

本题考查指令系统和计算机体系结构基础知识。

CISC（Complex Instruction Set Computer，复杂指令集计算机）的基本思想是：进一步增强原有指令的功能，用更为复杂的新指令取代原先由软件子程序完成的功能，实现软件功能的硬件化，导致机器的指令系统越来越庞大而复杂。CISC 计算机一般所含的指令数目至少 300 条以上，有的甚至超过 500 条。

RISC（Reduced Instruction Set Computer，精简指令集计算机）的基本思想是：通过减少指令总数和简化指令功能，降低硬件设计的复杂度，使指令能单周期执行，并通过优化编译提高指令的执行速度，采用硬布线控制逻辑优化编译程序。在 20 世纪 70 年代末开始兴起，导致机器的指令系统进一步精炼而简单。

参考答案

（2）A

试题（3）、（4）

浮点数的一般表示形式为 $N=2^E\times F$，其中 E 为阶码，F 为尾数。以下关于浮点表示的叙述中，错误的是__（3）__。两个浮点数进行相加运算，应首先__（4）__。

（3）A．阶码的长度决定浮点表示的范围，尾数的长度决定浮点表示的精度

B．工业标准 IEEE754 浮点数格式中阶码采用移码、尾数采用原码表示

C．规格化指的是阶码采用移码、尾数采用补码

D．规格化表示要求将尾数的绝对值限定在区间[0.5, 1)

（4）A．将较大的数进行规格化处理　　B．将较小的数进行规格化处理

C．将这两个数的尾数相加　　D．统一这两个数的阶码

试题（3）、（4）分析

本题考查数据表示基础知识。

为了提高运算的精度，需要充分地利用尾数的有效数位，通常采取浮点数规格化形式，即规定尾数的最高数位必须是一个有效值，即 $1/2 \leqslant |F| < 1$。在尾数用补码表示时，规格化浮点数应满足尾数最高数位与符号位不同，即当 $1/2 \leqslant F < 1$ 时，应有 0.1xx…x 形式；当 $-1 \leqslant M < -1/2$ 时，应有 1.0xx…x 形式。

需要注意的是，当 $M=-1/2$ 时，对于原码来说是规格化数，而对于补码来说不是规格化数。

两个浮点数进行相加运算时，首先需要对阶（使它们的阶码一致），然后再进行尾数的相加处理。

参考答案

（3）C　　（4）D

试题（5）

以下关于校验码的叙述中，正确的是__（5）__。

（5）A. 海明码利用多组数位的奇偶性来检错和纠错
B. 海明码的码距必须大于等于 1
C. 循环冗余校验码具有很强的检错和纠错能力
D. 循环冗余校验码的码距必定为 1

试题（5）分析

本题考查校验码基础知识。

一个编码系统中任意两个合法编码（码字）之间不同的二进数位数称为这两个码字的码距，而整个编码系统中任意两个码字的最小距离就是该编码系统的码距。为了使一个系统能检查和纠正一个差错，码间最小距离必须至少是 3。

海明码是一种可以纠正一位差错的编码，是利用奇偶性来检错和纠错的校验方法。海明码的基本意思是给传输的数据增加 r 个校验位，从而增加两个合法消息（合法码字）的不同位的个数（海明距离）。假设要传输的信息有 m 位，则经海明编码的码字就有 $n=m+r$ 位。

循环冗余校验码（CRC）编码方法是在 k 位信息码后再拼接 r 位的校验码，形成长度为 n 位的编码，其特点是检错能力极强且开销小，易于用编码器及检测电路实现。

在数据通信与网络中，通常 k 相当大，由一千甚至数千数据位构成一帧，而后采用 CRC 码产生 r 位的校验位。它只能检测出错误，而不能纠正错误。一般取 r=16，标准的 16 位生成多项式有 CRC–16＝$x^{16}+x^{15}+x^{2}+1$ 和 CRC–CCITT＝$x^{16}+x^{12}+x^{5}+1$。一般情况下，r 位生成多项式产生的 CRC 码可检测出所有的双错、奇数位错和突发长度小于等于 r 的突发错。用于纠错目的的循环码的译码算法比较复杂。

参考答案

（5）A

试题（6）

以下关于 Cache 的叙述中，正确的是__（6）__。

（6）A. 在容量确定的情况下，替换算法的时间复杂度是影响 Cache 命中率的关键因素
B. Cache 的设计思想是在合理成本下提高命中率
C. Cache 的设计目标是容量尽可能与主存容量相等
D. CPU 中的 Cache 容量应大于 CPU 之外的 Cache 容量

试题（6）分析

本题考查高速缓存基础知识。

Cache 是一个高速小容量的临时存储器，可以用高速的静态存储器（SRAM）芯片实现，可以集成到 CPU 芯片内部，或者设置在 CPU 与内存之间，用于存储 CPU 最经常访问的指令或者操作数据。Cache 的出现是基于两种因素：首先是由于 CPU 的速度和性能提高很快而主存速度较低且价格高，其次是程序执行的局部性特点。因此，才将速度比较快而容量有限的 SRAM 构成 Cache，目的在于尽可能发挥 CPU 的高速度。很显然，

要尽可能发挥 CPU 的高速度，就必须用硬件实现其全部功能。

参考答案

（6）B

试题（7）

网络安全体系设计可从物理线路安全、网络安全、系统安全、应用安全等方面来进行，其中，数据库容灾属于__（7）__。

（7）A．物理线路安全和网络安全　　B．应用安全和网络安全

C．系统安全和网络安全　　D．系统安全和应用安全

试题（7）分析

网络安全体系设计是逻辑设计工作的重要内容之一，数据库容灾属于系统安全和应用安全考虑范畴。

参考答案

（7）D

试题（8）

包过滤防火墙对数据包的过滤依据不包括__（8）__。

（8）A．源 IP 地址　　B．源端口号　　C．MAC 地址　　D．目的 IP 地址

试题（8）分析

本题考查防火墙相关知识。

包过滤防火墙对数据包的过滤依据包括源 IP 地址、源端口号、目标 IP 地址和目标端口号。

参考答案

（8）C

试题（9）

某网站向 CA 申请了数字证书，用户通过__（9）__来验证网站的真伪。

（9）A．CA 的签名　　B．证书中的公钥　　C．网站的私钥　　D．用户的公钥

试题（9）分析

本题考查数字证书相关知识点。

数字证书是由权威机构——CA 证书授权（Certificate Authority）中心发行的，能提供在 Internet 上进行身份验证的一种权威性电子文档，人们可以在因特网交往中用它来证明自己的身份和识别对方的身份。

数字证书包含版本、序列号、签名算法标识符、签发人姓名、有效期、主体名和主体公钥信息等并附有 CA 的签名，用户获取网站的数字证书后通过验证 CA 的签名来确认数字证书的有效性，从而验证网站的真伪。

在用户与网站进行安全通信时，用户发送数据时使用网站的公钥（从数字证书中获得）加密，收到数据时使用网站的公钥验证网站的数字签名，网站利用自身的私钥对发

送的消息签名和对收到的消息解密。

参考答案

（9）A

试题（10）

下列智力成果中，能取得专利权的是__（10）__。

（10）A. 计算机程序代码　　B. 游戏的规则和方法
　　C. 计算机算法　　D. 用于控制测试过程的程序

试题（10）分析

专利法明确规定，智力活动的规则和方法不取得专利权。智力活动规则和方法是直接作用于人的思维，与产业生产中的技术活动不发生直接关系，并不是对自然规律的利用，不属于技术方案。通常，智力活动规则和方法是一些人为的规则，如管理规则、游戏规则、统计方法、分类方法和计算方法等。

算法被认为是一种数学公式，反映的是自然法则和逻辑思维过程，属于智力活动规则和方法，不能受到专利法的保护。算法对于计算机软件而言极其重要，是计算机程序的基础，计算机软件总是离不开算法。因此，计算机软件本身属于智力活动规则和方法的范畴，不能够授予专利权。无论是源程序还是目标程序，它体现的是智力活动的规则和方法。因此仅以单纯的计算机程序为主题的发明创造是不能获得专利权的。

随着计算机技术的发展，不能授予专利权的智力活动规则和方法与能够授予专利权的方法之间的界线比较模糊了。例如，抽象出一种生产过程的数学模式，设计一种算法，通过予以运行用于控制该生产过程就能够获得专利保护。也就是说，智力活动规则和方法本身不被授予专利权，但进行智力活动的机器设备、装置或者根据智力活动规则和方法而设计制造的仪器、装置等都可以获得专利保护。即软件和硬件或者工业产品结合并表现为机器、装置或者为达到某种结果表现为方法时，软件可成为专利保护的对象。

参考答案

（10）D

试题（11）

软件权利人与被许可方签订一份软件使用许可合同。若在该合同约定的时间和地域范围内，软件权利人不得再许可任何第三人以此相同的方法使用该项软件，但软件权利人可以自己使用，则该项许可使用是__（11）__。

（11）A. 独家许可使用　　B. 独占许可使用
　　C. 普通许可使用　　D. 部分许可使用

试题（11）分析

软件许可使用一般有独占许可使用、独家许可使用和普通许可使用三种形式。独占许可使用，许可的是专有使用权，实施独占许可使用后，软件著作权人不得将软件使用

权授予第三方，软件著作权人自己不能使用该软件；独家许可使用，许可的是专有使用权，实施独家许可使用后，软件著作权人不得将软件使用权授予第三方，软件著作权人自己可以使用该软件；普通许可使用，许可的是非专有使用权，实施普通许可使用后，软件著作权人可以将软件使用权授予第三方，软件著作权人自己可以使用该软件。

参考答案

（11）A

试题（12）

多媒体中的“媒体”有两重含义，一是指存储信息的实体；二是指表达与传递信息的载体。__（12）__是存储信息的实体。

（12）A．文字、图形、磁带、半导体存储器

B．磁盘、光盘、磁带、半导体存储器

C．文字、图形、图像、声音

D．声卡、磁带、半导体存储器

试题（12）分析

通常所说的“媒体（Media）”包括两重含义：一是指信息的物理载体，即存储和传递信息的实体，如手册、磁盘、光盘、磁带以及相关的播放设备等（本题只涉及存储信息）；二是指承载信息的载体，即信息的表现形式（或者说传播形式），如文字、声音、图像、动画和视频等，即 CCITT 定义的存储媒体和表示媒体。表示媒体又可以分为三种类型：视觉类媒体（如位图图像、矢量图形、图表、符号、视频和动画等）、听觉类媒体（如音响、语音和音乐等）和触觉类媒体（如点、位置跟踪，力反馈与运动反馈等）。视觉和听觉类媒体是信息传播的内容，触觉类媒体是实现人机交互的手段。

参考答案

（12）B

试题（13）

RGB8:8:8 表示一帧彩色图像的颜色数为__（13）__种。

（13）A．2^3　　B．2^8　　C．2^{24}　　D．2^{512}

试题（13）分析

本题考查多媒体基础知识（图像深度）。

图像深度是指存储每个像素所用的位数，也是用来度量图像分辨率的。像素深度确定彩色图像的每个像素可能有的颜色数，或者确定灰度图像的每个像素可能有的灰度级数。如一幅图像的图像深度为 b 位，则该图像的最多颜色数或灰度级为 2^b 种。显然，表示一个像素颜色的位数越多，它能表达的颜色数或灰度级就越多。例如，只有 1 个分量的单色图像，若每个像素有 8 位，则最大灰度数目为 2^8=256；一幅彩色图像的每个像素用 R、G、B 三个分量表示，若 3 个分量的像素位数分别为 4、4、2，则最大颜色数目为 2^{4+4+2}=2^{10}=1024，就是说像素的深度为 10 位，每个像素可以是 2^{10} 种颜色中的一种。表

示一个像素的位数越多，它能表达的颜色数目就越多，它的深度就越深。

参考答案

（13）C

试题（14）

位图与矢量图相比，位图___（14）___。

（14）A．占用空间较大，处理侧重于获取和复制，显示速度快

B．占用空间较小，处理侧重于绘制和创建，显示速度较慢

C．占用空间较大，处理侧重于获取和复制，显示速度较慢

D．占用空间较小，处理侧重于绘制和创建，显示速度快

试题（14）分析

矢量图形是用一系列计算机指令来描述和记录一幅图的内容，即通过指令描述构成一幅图的所有直线、曲线、圆、圆弧、矩形等图元的位置、维数和形状，也可以用更为复杂的形式表示图像中曲面、光照和材质等效果。矢量图法实质上是用数学的方式（算法和特征）来描述一幅图形图像，在处理图形图像时根据图元对应的数学表达式进行编辑和处理。在屏幕上显示一幅图形图像时，首先要解释这些指令，然后将描述图形图像的指令转换成屏幕上显示的形状和颜色。编辑矢量图的软件通常称为绘图软件，如适于绘制机械图、电路图的AutoCAD软件等。这种软件可以产生和操作矢量图的各个成分，并对矢量图形进行移动、缩放、叠加、旋转和扭曲等变换。编辑图形时将指令转变成屏幕上所显示的形状和颜色，显示时也往往能看到绘图的过程。由于所有的矢量图形部分都可以用数学的方法加以描述，从而使得计算机可以对其进行任意放大、缩小、旋转、变形、扭曲、移动和叠加等变换，而不会破坏图像的画面。但是，用矢量图形格式表示复杂图像（如人物、风景照片），并且要求很高时，将需要花费大量的时间进行变换、着色和处理光照效果等。因此，矢量图形主要用于表示线框型的图画、工程制图和美术字等。

位图图像是指用像素点来描述的图。图像一般是用摄像机或扫描仪等输入设备捕捉实际场景画面，离散化为空间、亮度、颜色（灰度）的序列值，即把一幅彩色图或灰度图分成许许多多的像素（点），每个像素用若干二进制位来指定该像素的颜色、亮度和属性。位图图像在计算机内存中由一组二进制位组成，这些位定义图像中每个像素点的颜色和亮度。图像适合于表现比较细腻，层次较多，色彩较丰富，包含大量细节的图像，并可直接、快速地在屏幕上显示出来。但占用存储空间较大，一般需要进行数据压缩。

参考答案

（14）A

试题（15）

在采用结构化方法进行系统分析时，根据分解与抽象的原则，按照系统中数据处理的流程，用___（15）___来建立系统的逻辑模型，从而完成分析工作。

（15）A．E-R 图　　B．数据流图
C．程序流程图　　D．软件体系结构

试题（15）分析

本题考查结构化分析方法中图形工具的作用。数据流图摆脱系统的物理内容，在逻辑上描述系统的功能、输入、输出和数据存储等，是系统逻辑模型的重要组成部分。

参考答案

（15）B

试题（16）

面向对象开发方法的基本思想是尽可能按照人类认识客观世界的方法来分析和解决问题，__（16）__方法不属于面向对象方法。

（16）A．Booch　B．Coad　C．OMT　D．Jackson

试题（16）分析

本题考查面向对象开发方法。面向对象开发方法有 Booch 方法、Coad 方法和 OMT 方法。Jackson 方法是一种面向数据结构的开发方法。

参考答案

（16）D

试题（17）

确定构建软件系统所需要的人数时，无须考虑__（17）__。

（17）A．系统的市场前景　　B．系统的规模
C．系统的技术复杂性　　D．项目计划

试题（17）分析

本题考查项目管理内容。在对软件开发资源进行规划时，为了确定构建软件系统所需的人数，需要考虑软件系统的规模、系统的技术复杂性、项目计划和开发人员的技术背景等方面，而与系统是否有市场前景无关。

参考答案

（17）A

试题（18）

一个项目为了修正一个错误而进行了变更。但这个错误被修正后，却引起以前可以正确运行的代码出错。__（18）__最可能发现这一问题。

（18）A．单元测试　　B．接受测试
C．回归测试　　D．安装测试

试题（18）分析

本题考查软件测试知识。回归测试是在软件发生变更之后进行的测试，以发现在变更时可能引起的其他错误。

参考答案

（18）C

试题（19）

风险预测从两个方面评估风险，即风险发生的可能性以及__（19）__。

（19）A．风险产生的原因　　B．风险监控技术

C．风险能否消除　　D．风险发生所产生的后果

试题（19）分析

本题考查风险预测知识。风险预测从风险发生的可能性大小以及风险发生所产生的后果是否严重两个方面评估风险。

参考答案

（19）D

试题（20）

许多程序设计语言规定，程序中的数据都必须具有类型，其作用不包括__（20）__。

（20）A．便于为数据合理分配存储单元

B．便于对参与表达式计算的数据对象进行检查

C．便于定义动态数据结构

D．便于规定数据对象的取值范围及能够进行的运算

试题（20）分析

本题考查程序语言基础知识。

数据具有类型，便于编译程序在基础机器中完成对值的布局，同时还可用于检查表达式中对运算的应用是否正确。

参考答案

（20）C

试题（21）

以下关于 C/C++语言指针变量的叙述中，正确的是__（21）__。

（21）A．指针变量可以是全局变量也可以是局部变量

B．必须为指针变量与指针所指向的变量分配相同大小的存储空间

C．对指针变量进行算术运算是没有意义的

D．指针变量必须由动态产生的数据对象来赋值

试题（21）分析

本题考查程序语言基础知识。

变量是内存单元的抽象，用于在程序中表示数据。当变量存储的是内存单元地址时，称为指针变量，或者说指针变量指向了另一个变量。指针变量可以定义在函数或复合语句内，也可以定义在所有的函数之外，即可以是全局变量，也可以是局部变量。需要区分指针变量与指针所指向的变量，无论指针变量指向何种变量，其存储空间大小都是一

样的。当指针变量指向数组中的一个元素时，对指针变量进行算术运算可以使其指向同一个数组中的其他元素。

参考答案

（21）A

试题（22）

将高级语言源程序翻译为机器语言程序的过程中常引入中间代码。以下关于中间代码的叙述中，错误的是__（22）__。

（22）A. 不同的高级程序语言可以产生同一种中间代码

B. 使用中间代码有利于进行与机器无关的优化处理

C. 使用中间代码有利于提高编译程序的可移植性

D. 中间代码与机器语言代码在指令结构上必须一致

试题（22）分析

本题考查程序语言基础知识。

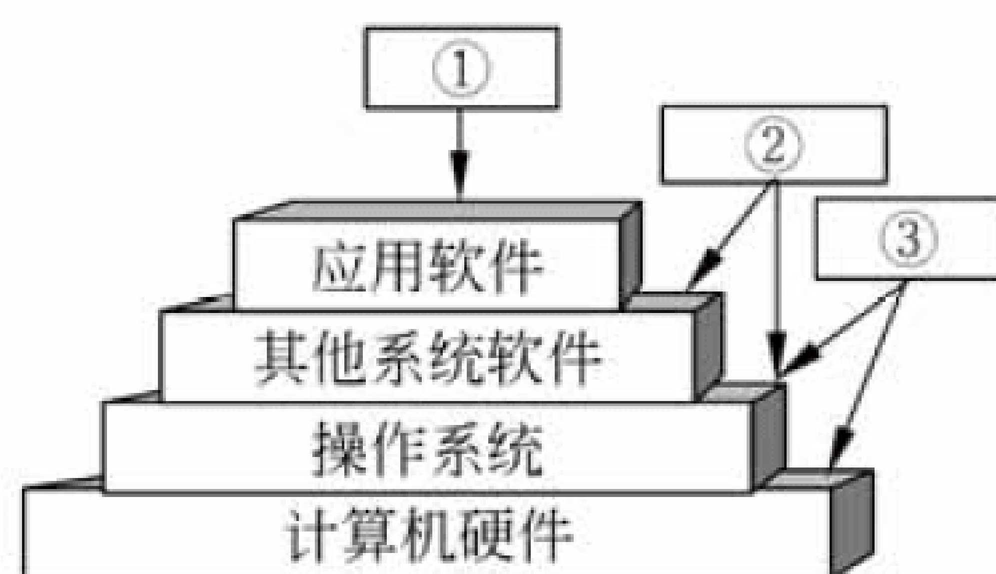
“中间代码”是一种简单且含义明确的记号系统，与具体的机器无关，可以有若干种形式。可以将不同的高级程序语言翻译成同一种中间代码。由于与具体机器无关，使用中间代码有利于进行与机器无关的优化处理，以及提高编译程序的可移植性。

参考答案

（22）D

试题（23）、（24）

操作系统是裸机上的第一层软件，其他系统软件（如__（23）__等）和应用软件都是建立在操作系统基础上的。下图①②③分别表示__（24）__。



（23）A. 编译程序、财务软件和数据库管理系统软件

B. 汇编程序、编译程序和 Java 解释器

C. 编译程序、数据库管理系统软件和汽车防盗程序

D. 语言处理程序、办公管理软件和气象预报软件

（24）A. 应用软件开发者、最终用户和系统软件开发者

B. 应用软件开发者、系统软件开发者和最终用户

C. 最终用户、系统软件开发者和应用软件开发者

D. 最终用户、应用软件开发者和系统软件开发者

试题（23）、（24）分析

本题考查操作系统基本概念。

财务软件、汽车防盗程序、办公管理软件和气象预报软件都属于应用软件，而选项A、C和D中含有这些软件。选项B中汇编程序、编译程序和数据库管理系统软件都属于系统软件。

计算机系统由硬件和软件两部分组成。通常把未配置软件的计算机称为裸机，直接使用裸机不仅不方便，而且将严重降低工作效率和机器的利用率。操作系统（Operating System）的目的是为了填补人与机器之间的鸿沟，即建立用户与计算机之间的接口而为裸机配置的一种系统软件。由图1可以看出，操作系统是裸机上的第一层软件，是对硬件系统功能的首次扩充。它在计算机系统中占据重要而特殊的地位，所有其他软件，如编辑程序、汇编程序、编译程序和数据库管理系统等系统软件，以及大量的应用软件都是建立在操作系统基础上的，并得到它的支持和取得它的服务。从用户角度看，当计算机配置了操作系统后，用户不再直接使用计算机系统硬件，而是利用操作系统所提供的命令和服务去操纵计算机，操作系统已成为现代计算机系统中必不可少的最重要的系统软件，因此把操作系统看作是用户与计算机之间的接口。因此，操作系统紧贴系统硬件之上，所有其他软件之下（是其他软件的共同环境）。

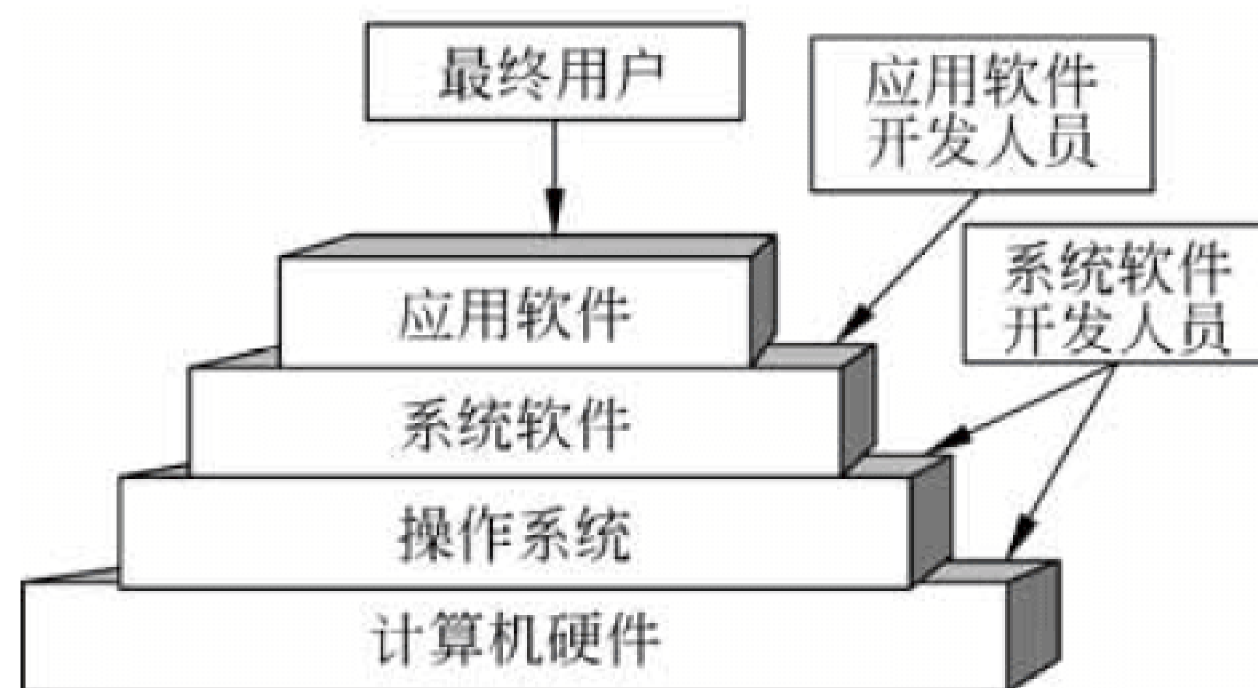


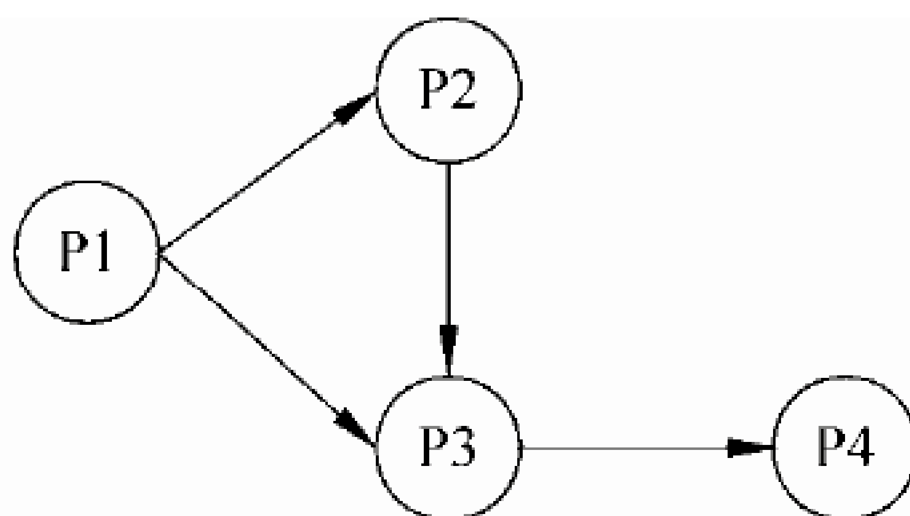
图 操作系统在计算机系统中的地位

参考答案

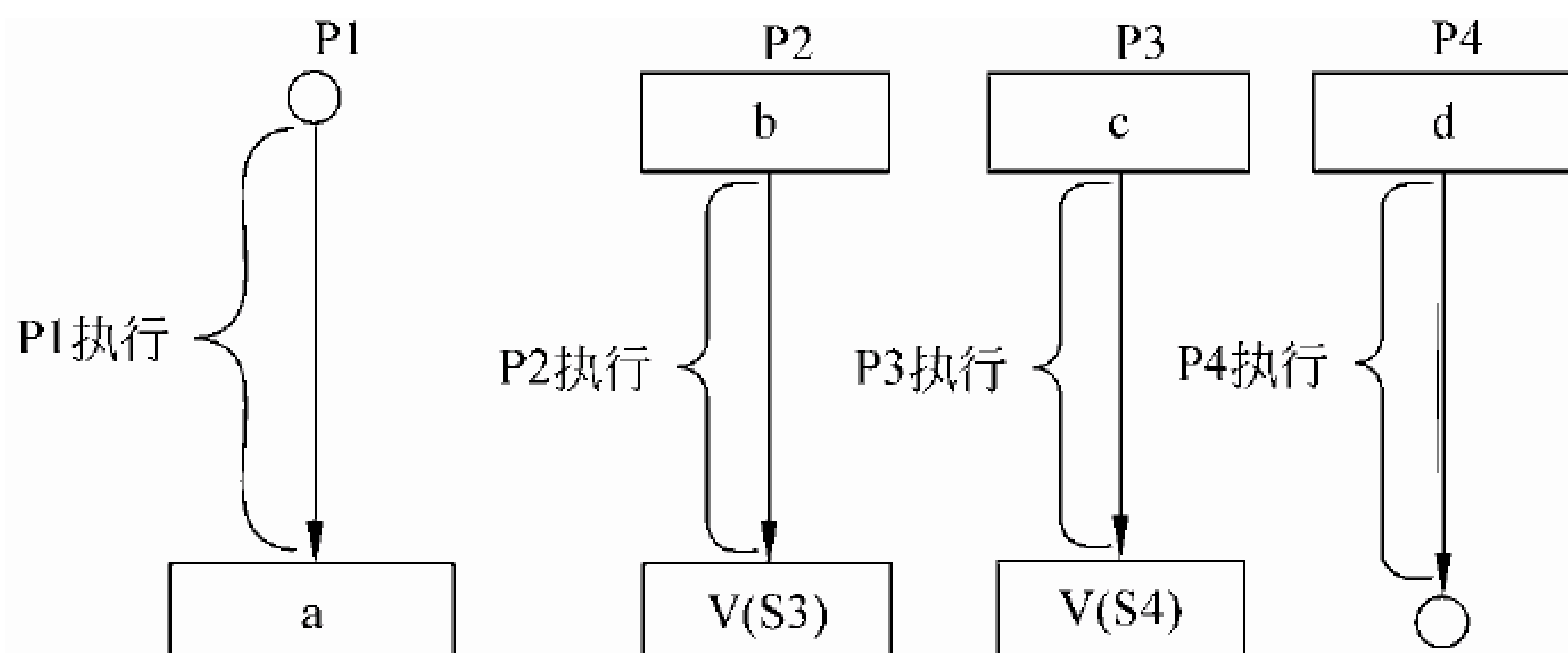
（23）B （24）D

试题（25）、（26）

进程P1、P2、P3和P4的前趋图如下：



若用 PV 操作控制这几个进程并发执行的过程，则需要设置 4 个信号量 S1、S2、S3 和 S4，且信号量初值都等于零。下图中 a 和 b 应分别填写__（25）__，c 和 d 应分别填写__（26）__。



（25）A. P（S1）P（S2）和 P（S3）　　B. P（S1）P（S2）和 V（S1）
　　C. V（S1）V（S2）和 P（S1）　　D. V（S1）V（S2）和 V（S3）

（26）A. P（S1）P（S2）和 P（S4）　　B. P（S2）P（S3）和 P（S4）
　　C. V（S1）V（S2）和 V（S4）　　D. V（S2）V（S3）和 V（S4）

试题（25）、（26）分析

本题考查操作系统进程管理中 PV 操作方面的基本知识。

根据题意，进程 P2、P3 等待 P1 的结果，因此当 P1 执行完毕需要使用 V 操作通知 P2、P3，即 a 处填 V（S1）V（S2），b 处应填 P（S1）。

根据题意，进程 P3 要执行需要测试 P1、P2 有没有消息，故应该在 c 处填 P（S2）、P（S3）。当 P3 执行完毕需要使用 V 操作通知 P4，即在 d 处填 P（S4）。

参考答案

（25）C　　（26）B

试题（27）

若系统正在将__（27）__文件修改的结果写回磁盘时系统发生崩溃，则对系统的影响相对较大。

（27）A. 空闲块　　B. 目录　　C. 用户数据　　D. 用户程序

试题（27）分析

本题考查操作系统文件管理可靠性方面的基础知识。

影响文件系统可靠性因素之一是文件系统的一致性问题。很多文件系统是先读取磁盘块到主存，在主存进行修改，修改完毕再写回磁盘。例如读取某磁盘块，修改后再将信息写回磁盘前系统崩溃，则文件系统就可能会出现不一致性状态。如果这些未被写回的磁盘块是索引节点块、目录块或空闲块，特别是系统目录文件，那么对系统的影响相对较大，且后果也是不堪设想的。通常解决方案是采用文件系统的一致性检查，一致性检查包括块的一致性检查和文件的一致性检查。

参考答案

（27）B

试题（28）

UNIX 系统采用直接、一级、二级和三级间接索引技术访问文件，其索引结点有 13 个地址项（i_addr[0]～i_addr[12]）。如果每个盘块的大小为 1KB，每个盘块号占 4B，则进程 A 访问文件 F 中第 11264 字节处的数据时，__(28)__。

（28）A．可直接寻址　　B．需要一级间接寻址

C．需要二级间接寻址　　D．需要三级间接寻址

试题（28）分析

本题考查 UNIX 文件系统管理中目录结构方面的基础知识。

UNIX 系统采用直接、一级、二级和三级间接索引技术访问文件，其索引结点有 13 个地址项（i_addr[0]～i_addr[12]）。其中 i_addr[0]～i_addr[9]采用直接索引技术访问文件，i_addr[10] 采用一级间接索引技术访问文件，i_addr[11] 采用二级间接索引技术访问文件，i_addr[12] 采用三级间接索引技术访问文件。如下图所示。

根据题意，每个盘块的大小为 1KB，每个盘块号占 4B，那么，一个盘块可以存放 256 个盘块号。又因为进程 A 访问文件 F 中第 11264 字节处的数据，该数据应该放在 11264/1024=11 号逻辑盘块中，从上图中可以看出 11 号逻辑盘块应采用一级间接索引。

参考答案

（28）B

试题（29）

软件能力成熟度模型（CMM）的第 4 级（已管理级）的核心是__（29）__。

（29）A．建立基本的项目管理和实践来跟踪项目费用、进度和功能特性

B．组织具有标准软件过程

C．对软件过程和产品都有定量的理解和控制

D．先进的新思想和新技术促进过程不断改进

试题（29）分析

本题考查成熟度等级（CMM）知识。在 CMM 的不同等级有不同的核心。在可重复级，建立了基本的项目管理过程和实践来跟踪项目费用、进度和功能特性。在已定义级，所有项目都采用根据实际情况修改后得到的标准软件过程来开发和维护软件。在已管理级，收集对软件过程和产品质量的详细度量，对软件过程和产品都有定量的理解与控制。在优化级，过程的量化反馈和先进的新思想、新技术促使过程不断改进。

参考答案

（29）C

试题（30）

软件系统设计的主要目的是为系统制定蓝图，__（30）__并不是软件设计模型所关注的。

（30）A．系统总体结构　B．数据结构　C．界面模型　D．项目范围

试题（30）分析

本题考查软件系统设计知识。系统设计为系统制定蓝图，软件设计模型关注新系统总体结构、代码设计、处理过程、数据结构和界面模型等。对项目范围的关注在软件设计之前。

参考答案

（30）D

试题（31）

ISO/IEC 9126 软件质量模型中，可靠性质量特性包括多个子特性。一软件在故障发生后，要求在 90 秒内恢复其性能和受影响的数据，与达到此目的有关的软件属性为__（31）__子特性。

（31）A．容错性　B．成熟性　C．易恢复性　D．易操作性

试题（31）分析

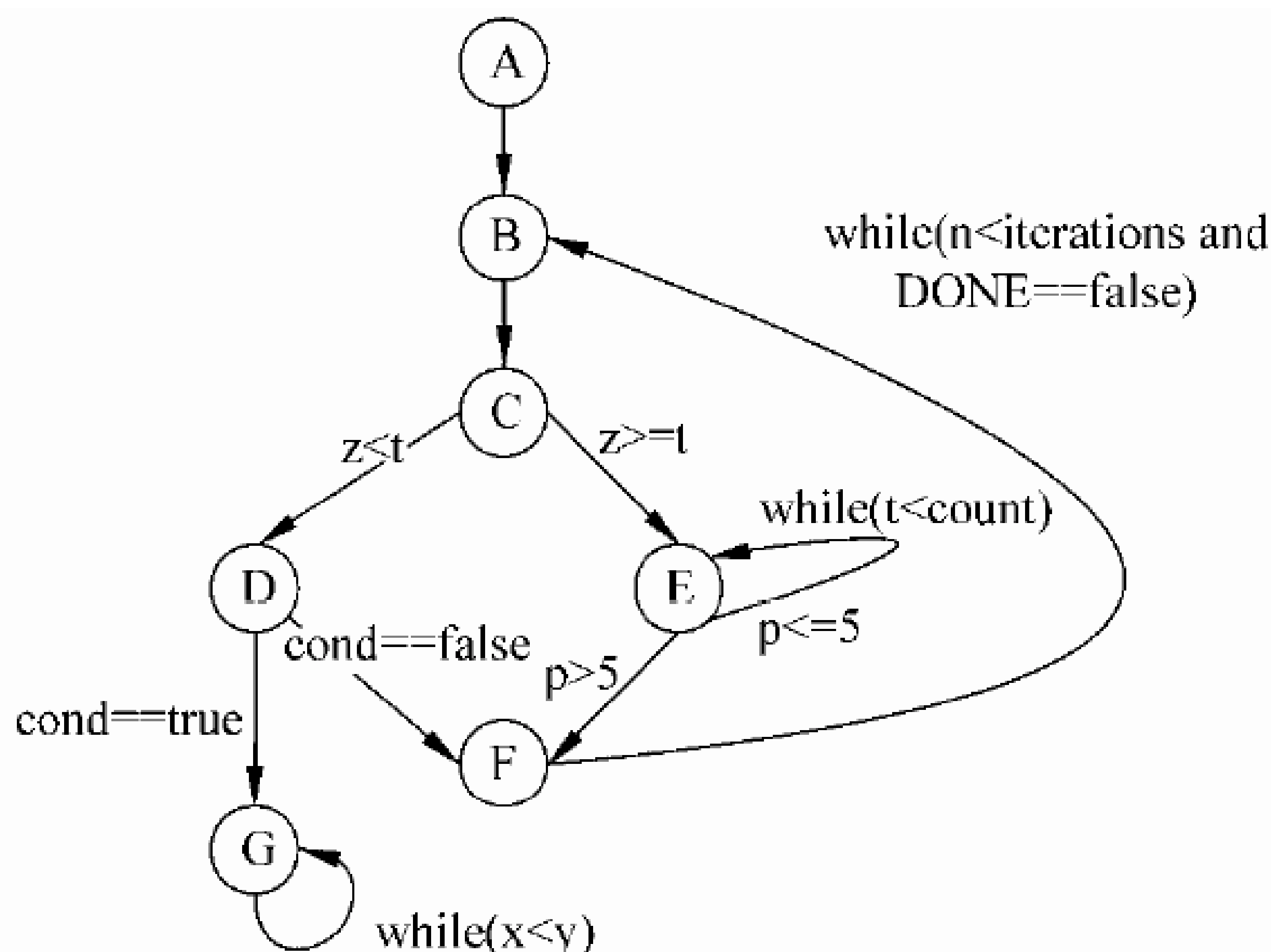
本题考查软件质量模型知识。ISO/IEC 9126 软件质量模型中可靠性质量特性是指在规定的一段时间内和规定的条件下，软件维护其性能水平有关的能力。包括的子特性有成熟性、容错性和易恢复性。其中易恢复性是在故障发生后，重新建立其性能水平并恢复直接受影响数据的能力，以及为达到此目的所需的时间和工作有关的软件属性。软件故障发生后，要在 90 秒内恢复其性能和受影响的数据，达到这一目的有关的属性即为易恢复性子特性。

参考答案

（31）C

试题（32）

某程序的程序图如下所示，运用 McCabe 度量法对其进行度量，其环路复杂度是＿（32）＿。



（32）A．2　　B．3　　C．4　　D．5

试题（32）分析

本题考查 McCabe 度量法。McCabe 度量法是通过定义环路复杂度，建立程序复杂性的度量，它基于一个程序模块的程序图中环路的个数。计算有向图 G 的环路复杂性的公式为 $V(G)=m-n+2$，其中 $V(G)$是有向图 G 中的环路个数，m 是 G 中的有向弧数，n 是 G 中的节点数。要采用 McCabe 度量法度量程序复杂度，需要先画出程序图，识别有向图中节点数 n=7 和有向弧数 m=10，然后利用环路复杂性计算公式 $V(G)=m-n+2$ 进行计算，得到图中程序图的程序复杂度是 5。

参考答案

（32）D

试题（33）

系统开发计划用于系统开发人员与项目管理人员在项目期内进行沟通，它包括＿（33）＿

和预算分配表等。

（33）A. PERT 图　　B. 总体规划　　C. 测试计划　　D. 开发合同

试题（33）分析

本题考查系统开发计划文档知识。

用于系统开发人员与项目管理人员在项目期内进行沟通的文档主要有系统开发计划，包括工作任务分解表、PERT 图、甘特图和预算分配表等。总体规划和开发合同用于与系统分析人员在系统规划和系统分析阶段的沟通。测试计划用于系统测试人员与系统开发人员之间的沟通。

参考答案

（33）A

试题（34）

改正在软件系统开发阶段已经发生而系统测试阶段还没有发现的错误，属于＿（34）＿维护。

（34）A. 正确性　　B. 适应性　　C. 完善性　　D. 预防性

试题（34）分析

本题考查软件维护知识。

软件维护的内容一般包括准确性维护、适应性维护、完善性维护和预防性维护。正确性维护是指改正在系统开发阶段已发生而系统测试阶段尚未发现的错误。适应性维护是指使应用软件适应信息技术变化和管理需求变化而进行的修改。完善性维护主要是指对已有的软件系统增加一些在系统分析和设计阶段中没有规定的功能与性能特征。预防性维护是指为了适应未来的软硬件环境的变化，主动增加预防性的新功能，以使应用系统适应各类变化而不被淘汰。

参考答案

（34）A

试题（35）

某系统重用了第三方组件（但无法获得其源代码），则应采用＿（35）＿对组件进行测试。

（35）A. 基本路径覆盖　　B. 分支覆盖　　C. 环路覆盖　　D. 黑盒测试

试题（35）分析

本题考查软件动态测试方法。

软件测试方法分为静态测试和动态测试。静态测试是被测试程序不在机器上运行而采用人工检测和计算机辅助静态分析手段对程序进行检测。动态测试是通过运行程序发现错误。对软件产品进行动态测试时可以采用黑盒测试和白盒测试法。黑盒测试法在完全不考虑软件的内部结构和特性的情况下，测试软件的外部特性。对于无法获得源代码的第三方组件，可采用黑盒对组件进行测试。基本路径覆盖、分支覆盖和环路覆盖需要根据程序的内部结构和逻辑进行测试。

参考答案

（35）D

试题（36）

极限编程（XP）由价值观、原则、实践和行为四个部分组成，其中价值观包括沟通、简单性、__（36）__。

（36）A. 好的计划　B. 不断的发布　C. 反馈和勇气　D. 持续集成

试题（36）分析

本题考查敏捷开发方法。

极限编程（XP）是敏捷开发的典型方法之一，是一种轻量级（敏捷）、高效、低风险、柔性、可预测的、科学的软件开发方法，它由价值观、原则、实践和行为 4 个部分组成。其中 4 大价值观为沟通、简单性、反馈和勇气。

参考答案

（36）C

试题（37）

以下关于类和对象的叙述中，错误的是__（37）__。

（37）A. 类是具有相同属性和服务的一组对象的集合
B. 类是一个对象模板，用它仅可以产生一个对象
C. 在客观世界中实际存在的是类的实例，即对象
D. 类为属于该类的全部对象提供了统一的抽象描述

试题（37）分析

本题考查面向对象中类和对象的基本概念。

面向对象中，类是对象的模板，描述了一组对象共有的成员属性和方法。类是一组对象的抽象描述，通过类可以产生一个或多个对象，这些对象具有相同的属性和方法。

参考答案

（37）B

试题（38）～（40）

__（38）__是把对象的属性和服务结合成一个独立的系统单元，并尽可能隐藏对象的内部细节；__（39）__是指子类可以自动拥有父类的全部属性和服务；__（40）__是对象发出的服务请求，一般包含提供服务的对象标识、服务标识、输入信息和应答信息等。

（38）A. 继承　B. 多态　C. 消息　D. 封装
（39）A. 继承　B. 多态　C. 消息　D. 封装
（40）A. 继承　B. 多态　C. 消息　D. 封装

试题（38）～（40）分析

本题考查面向对象的基本概念。

封装的作用是把属性和服务结合成一个独立的系统单元，并隐藏对象的内部细节；而继承则使得子类可以自动拥有父类的全部属性和服务；消息则是对象发出的服务请求；

多态是指面向对象中由重载或过载机制实现的结果。

参考答案

（38）D （39）A （40）C

试题（41）

以下关于面向对象分析的叙述中，错误的是__（41）__。

（41）A．面向对象分析着重分析问题域和系统责任

B．面向对象分析需要考虑系统的测试问题

C．面向对象分析忽略与系统实现有关的问题

D．面向对象分析建立独立于实现的系统分析模型

试题（41）分析

本题考查面向对象分析的基本概念。

面向对象分析主要回答软件系统需要解决什么问题，在面向对象分析阶段，并不考虑系统实现以及系统的测试问题，强调建立独立与系统实现的系统分析模型，其关注点仍然侧重于问题域。

参考答案

（41）B

试题（42）

以下关于面向对象设计的叙述中，错误的是__（42）__。

（42）A．高层模块不应该依赖于底层模块

B．抽象不应该依赖于细节

C．细节可以依赖于抽象

D．高层模块无法不依赖于底层模块

试题（42）分析

本题考查面向对象设计的基本原则。

面向对象设计中存在很多原则，其中高层模块不应该依赖于底层模块就是其中一条。传统的结构化设计中，一般的高层模块需要调用或使用底层模块，导致高层模块必须依赖于底层模块，但借助面向对象中的抽象类、接口等一系列机制，可以使得高层模块仅仅和接口耦合，而和实际的底层模块没有直接关联。

参考答案

（42）D

试题（43）～（45）

采用__（43）__设计模式可保证一个类仅有一个实例；采用__（44）__设计模式可将对象组合成树形结构以表示“部分-整体”的层次结构，使用户对单个对象和组合对象的使用具有一致性；采用__（45）__设计模式可动态地给一个对象添加一些额外的职责。

（43）A．命令（Command） B．单例（Singleton）

C．装饰（Decorate） D．组合（Composite）

（44）A. 命令（Command） B. 单例（Singleton）
C. 装饰（Decorate） D. 组合（Composite）
（45）A. 命令（Command） B. 单例（Singleton）
C. 装饰（Decorate） D. 组合（Composite）

试题（43）～（45）分析

本题考查面向对象设计中的设计模式。

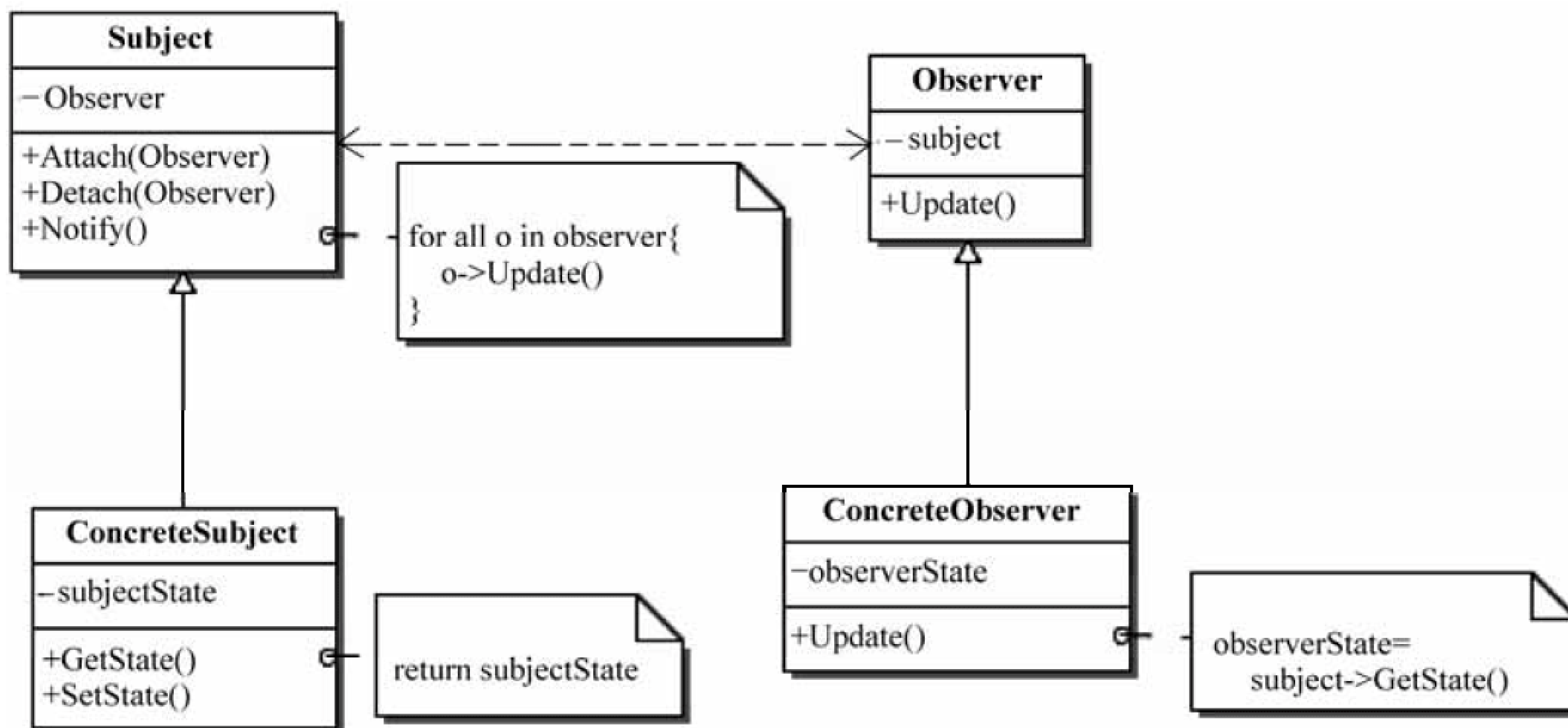
单例设计模式的目的就是保证一个类仅能够生成一个对象；组合设计模式则用于表示“部分-整体”的层次结构，并且对部分和整体的使用具有一致性；装饰设计模式则动态地给一个对象增加一些额外的职责，无须改变类的设计和实现。

参考答案

（43）B （44）D （45）C

试题（46）、（47）

下列 UML 类图表示的是__（46）__设计模式。该设计模式中，__（47）__。



（46）A. 备忘录（Memento） B. 策略（Strategy）
C. 状态（State） D. 观察者（Observer）
（47）A. 一个 Subject 对象可对应多个 Observer 对象
B. Subject 只能有一个 ConcreteSubject 子类
C. Observer 只能有一个 ConcreteObserver 子类
D. 一个 Subject 对象必须至少对应一个 Observer 对象

试题（46）、（47）分析

本题考查面向对象设计中的设计模式。

题中的图标是观察者设计模式，在该设计模式中，一个 Subject 类可以有多个 ConcreteSubject 子类，一个 Observer 类也可以有多个 ConcreteObserver 子类；一个 Subject

对象可以对应多个 Observer 对象。

参考答案

（46）D　（47）A

试题（48）

以下关于编译系统对某高级语言进行翻译的叙述中，错误的是＿（48）＿。

（48）A．词法分析将把源程序看作一个线性字符序列进行分析

B．语法分析阶段可以发现程序中所有的语法错误

C．语义分析阶段可以发现程序中所有的语义错误

D．目标代码生成阶段的工作与目标机器的体系结构相关

试题（48）分析

本题考查程序语言翻译基础知识。

词法分析是编译过程的第一阶段，其任务是对源程序从前到后（从左到右）逐个字符地扫描，从中识别出一个个的“单词”符号。语法分析的任务是在词法分析的基础上，根据语言的语法规则将单词符号序列分解成各类语法单位，如“表达式”“语句”和“程序”等。语义分析阶段主要检查源程序是否包含语义错误，并收集类型信息供后面的代码生成阶段使用。只有语法和语义都正确的源程序才能被翻译成正确的目标代码。目标代码生成是编译器工作的最后一个阶段。这一阶段的任务是把中间代码变换成特定机器上的绝对指令代码、可重定位的指令代码或汇编指令代码，这个阶段的工作与具体的机器密切相关。

源程序不可避免地会有一些错误，这些错误大致可分为语法错误和语义错误。语法错误是指语言结构上的使用错误，是指编译时所发现的程序错误，如单词拼写错误、标点符号错、表达式中缺少操作数、括号不匹配等有关语言结构上的错误。

参考答案

（48）C

试题（49）

若一个程序语言可以提供链表的定义和运算，则其运行时的＿（49）＿。

（49）A．数据空间适合采用静态存储分配策略

B．数据空间必须采用堆存储分配策略

C．指令空间需要采用栈结构

D．指令代码必须放入堆区

试题（49）分析

本题考查程序语言基础知识。

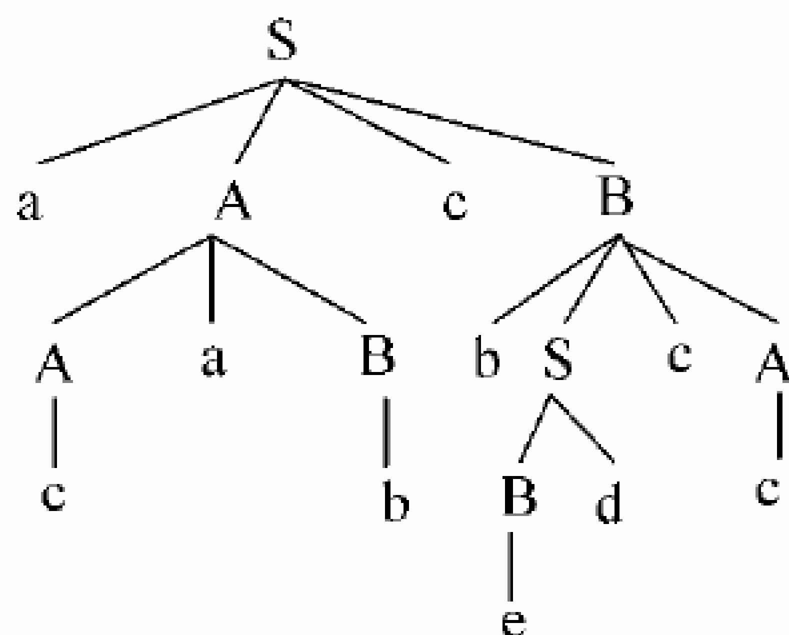
链表中的结点空间需要程序员根据需要申请和释放，因此，数据空间应采用堆存储分配策略。

参考答案

（49）B

试题（50）

由某上下文无关文法 M[S]推导出某句子的分析树如下图所示，则错误的叙述是<u>（50）</u>。



（50）A．该文法推导出的句子必须以“a”开头

B．acabcbdcc 是该文法推导出的一个句子

C．“S->aAcB”是该文法的一个产生式

D．a、b、c、d 属于该文法的终结符号集

试题（50）分析

本题考查程序语言基础知识。

推导就是从文法的开始符号 S 出发，反复使用产生式，将产生式左部的非终结符替换为右部的文法符号序列（展开产生式用⇒表示），直到产生一个终结符的序列时为止。从题中给出的分析树可得到如下的一个最左推导过程 S⇒aAcB⇒aAaBcB⇒acaBcB⇒acabcB⇒acabcbScA⇒acabcbBdcA⇒acabcbdcA⇒acabcbdcc，因此，acabcbdcc 是该文法推导出的一个句子。其中用到的产生式如下：

S->aAcB	S->Bd	
A->AaB	A->c	
B->bScA	B->b	B->ε

从起始符号 S 出发也可以如下推导：S⇒Bd⇒ εd⇒d，即该文法推导出的句子也可以为 d，因此选项 A 错误。

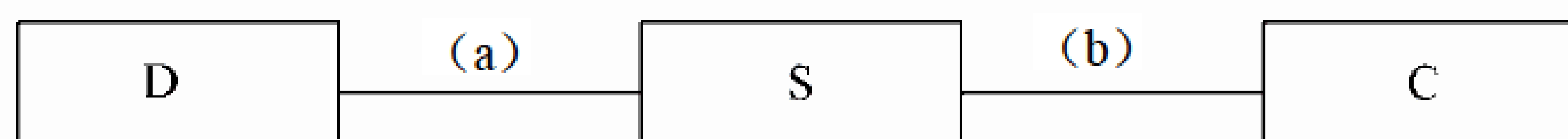
参考答案

（50）A

试题（51）、（52）

假设有学生 S（学号，姓名，性别，入学时间，联系方式），院系 D（院系号，院系名称，电话号码，负责人）和课程 C（课程号，课程名）三个实体，若一名学生属于一个院系，一个院系有多名学生；一名学生可以选择多门课程，一门课程可被多名学生选择，则图中（a）和（b）分别为<u>（51）</u>联系。假设一对多联系不转换为一个独立的关系

模式，那么生成的关系模式＿（52）＿。



（51）A. 1 * 和 1 *　　B. 1 * 和 * 1
　　　C. 1 * 和 * *　　D. * 1 和 * *

（52）A. S 中应加入关系模式 D 的主键
　　　B. S 中应加入关系模式 C 的主键
　　　C. D 中应加入关系模式 S 的主键
　　　D. C 中应加入关系模式 S 的主键

试题（51）、（52）分析

本题考查的是应试者对数据库系统中 E-R 模型方面的基础知识。

根据题意，学生 S 关系中一个学生属于一个院系，一个院系有多个学生，因此学生 S 和部门院系 D 之间是一个一对多的联系，故（a）应为 1 * ；又因为一个学生可以选择多门课程，一门课程可由多名学生选择，所以学生 S 和课程 C 关系之间是一个多对多的联系，故（b）应为 * * 。

假设一对多联系（a）不转换为一个独立的关系模式，那么按照 E-R 模型向关系模型转换的规则，应当将一端的主键并入多端。所以，生成的关系模式学生 S 中应加入关系模式 D 的主键。

参考答案

（51）C　（52）A

试题（53）

软硬件故障常造成数据库中的数据破坏。数据库恢复就是＿（53）＿。

（53）A. 重新安装数据库管理系统和应用程序
　　　B. 重新安装应用程序，并将数据库做镜像
　　　C. 重新安装数据库管理系统，并将数据库做镜像
　　　D. 在尽可能短的时间内，把数据库恢复到故障发生前的状态

试题（53）分析

本题考查关系数据库事务处理方面的基础知识。

为了保证数据库中数据的安全可靠和正确有效，数据库管理系统（DBMS）提供数据库恢复、并发控制、数据完整性保护与数据安全性保护等功能。数据库在运行过程中由于软硬件故障可能造成数据被破坏，数据库恢复就是在尽可能短的时间内把数据库恢复到故障发生前的状态。

参考答案

（53）D

试题（54）～（56）

设有员工实体 Emp（员工号，姓名，性别，年龄，出生年月，联系方式，部门号），其中“联系方式”要求记录该员工的手机号码和办公室电话，部门号要求参照另一部门实体 Dept 的主码“部门号”。Emp 实体中存在派生属性和多值属性：__（54）__；对属性部门号应该进行__（55）__约束；可以通过命令__（56）__修改表中的数据。

（54）A．年龄和出生年月　　B．年龄和联系方式
　　C．出生年月和联系方式　　D．出生年月和年龄

（55）A．非空主键　　B．主键
　　C．外键　　D．候选键

（56）A．INSERT　　B．DELETE　　C．UPDATE　　D．MODIFY

试题（54）～（56）分析

本题考查数据库系统设计的基本概念和 SQL 基本命令。

概念模式设计阶段是根据对用户信息需要的分析设计 E-R 图，对于属性的分析，派生属性是指可以由其他属性经过运算得到的属性，因而派生属性产生冗余，通常不存储，如员工实体 Emp 的年龄；多值属性是指一个实体在该属性上会同时取多个属性值，这些值也都必须存储，如员工实体 Emp 的联系方式。数据约束也是描述用户信息的，根据参照完整性约束的定义，Emp 实体中的部门号属性应为外码，应该用 Foreign Key 进行外键约束。

标准 SQL 对数据库对象（基本表、索引和视图等）的创建、修改和删除定义了一组操作分别为 CREATE、ALTER 和 DROP；对基本表和视图中数据的插入、删除和修改定义了操作分别为 INSERT、DELETE 和 UPDATE，这两组操作不能混淆。

参考答案

（54）B　　（55）C　　（56）C

试题（57）、（58）

已知一个二叉树的先序遍历序列为①、②、③、④、⑤，中序遍历序列为②、①、④、③、⑤，则该二叉树的后序遍历序列为__（57）__。对于任意一棵二叉树，叙述错误的是__（58）__。

（57）A．②、③、①、⑤、④
　　B．①、②、③、④、⑤
　　C．②、④、⑤、③、①
　　D．④、⑤、③、②、①

（58）A．由其后序遍历序列和中序遍历序列可以构造该二叉树的先序遍历序列

B. 由其先序遍历序列和后序遍历序列可以构造该二叉树的中序遍历序列

C. 由其层序遍历序列和中序遍历序列可以构造该二叉树的先序遍历序列

D. 由其层序遍历序列和后序遍历序列不能构造该二叉树的中序遍历序列

试题（57）、（58）分析

本题考查数据结构基础知识。

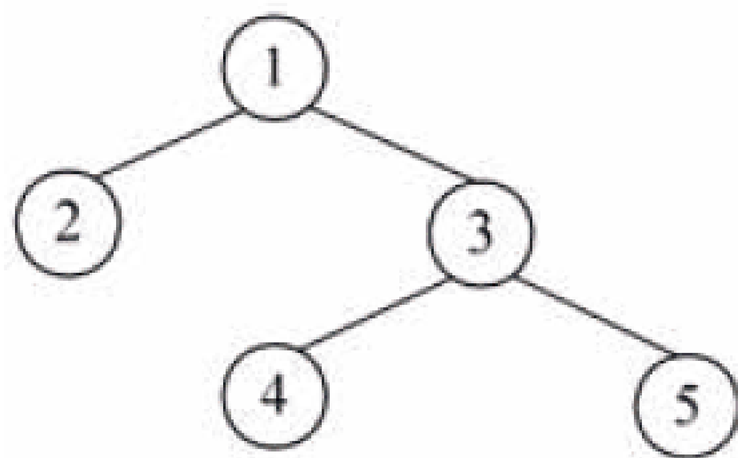
遍历运算是二叉树的基本运算，主要有先序、中序、后序和层序遍历。

先序遍历的基本方法：对于非空二叉树，先访问根结点，然后先序遍历根的左子树，最后先序遍历根的右子树。因此，若已知某二叉树的先序遍历序列，则可直接得到其树根结点。

中序遍历的基本方法：对于非空二叉树，先中序遍历根的左子树，然后访问根结点，最后中序遍历根的右子树。因此，若已知某二叉树的根结点，则可根据中序遍历序列将该二叉树左右子树上的结点划分开。

后序遍历的基本方法：对于非空二叉树，首先后序遍历根的左子树，接着后序遍历根的右子树，最后访问根结点。因此，若已知某二叉树的后序遍历序列，则可直接得到其树根结点。

题中给出的先序遍历序列为①、②、③、④、⑤，可知树根结点是①，据此再结合中序遍历序列②、①、④、③、⑤，可知②是根结点①左子树上的结点，由于是左子树上唯一的一个结点，因此②是根结点①的左孩子。对于右子树上的结点④、③、⑤，因右子树的先序遍历序列为③、④、⑤，因此③是根结点①的右孩子。依此类推，可知④是结点③的左孩子，⑤是结点③的右孩子。该二叉树如下图所示。



从二叉树的遍历过程可知，从先序遍历序列和后序遍历序列中无法将左子树和右子树上的结点区分开，因此，由某棵二叉树的先序遍历序列和后序遍历序列不能构造出该二叉树的中序遍历序列。

层序遍历二叉树的方法：设二叉树的根结点所在层数为 1，则层序遍历二叉树的操作定义为从树的根结点出发，首先访问第一层的结点（根结点），然后从左到右依次访问第二层上的结点，接着是第三层上的结点，依此类推，自上而下、自左至右逐层访问树中各层上的结点。

参考答案

（57）C　（58）B

试题（59）

邻接矩阵和邻接表是图（网）的两种基本存储结构，对于具有 n 个顶点、e 条边的图，（59）。

（59）A．进行深度优先遍历运算所消耗的时间与采用哪一种存储结构无关

B．进行广度优先遍历运算所消耗的时间与采用哪一种存储结构无关

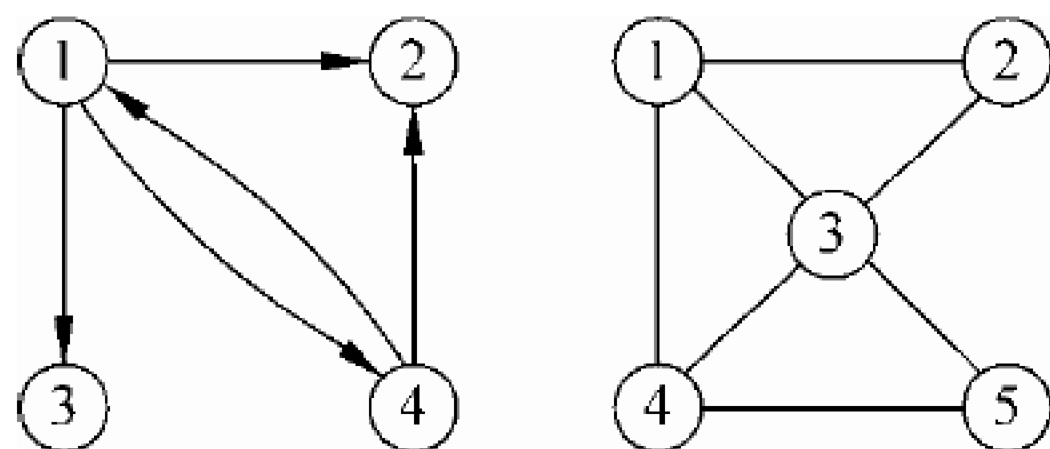
C．采用邻接表表示图时，查找所有顶点的邻接顶点的时间复杂度为 $O(n*e)$

D．采用邻接矩阵表示图时，查找所有顶点的邻接顶点的时间复杂度为 $O(n^2)$

试题（59）分析

本题考查数据结构基础知识。

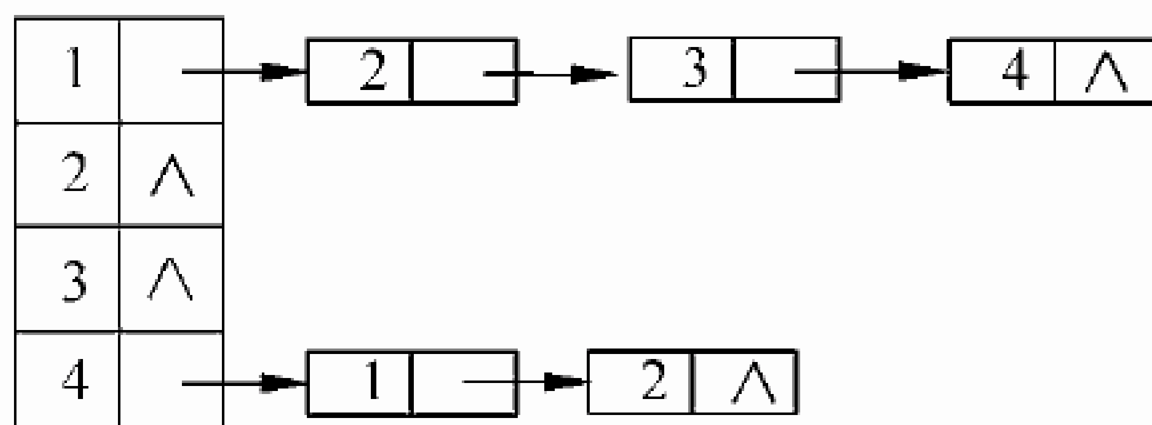
设某有向图和无向图如下所示。



下面的矩阵 A 是该有向图的邻接矩阵，B 为无向图的邻接矩阵。

$$A=\begin{bmatrix}0&1&1&1\\0&0&0&0\\0&0&0&0\\1&1&0&0\end{bmatrix}\qquad B=\begin{bmatrix}0&1&1&1&0\\1&0&1&0&0\\1&1&0&1&1\\1&0&1&0&1\\0&0&1&1&0\end{bmatrix}$$

上面有向图的邻接链表如下图所示。



图的遍历运算是按照某种策略访问图中的每一个顶点，实质上是通过边或弧找邻接点的过程，因此广度优先搜索遍历图和深度优先搜索遍历图的时间复杂度相同，其不同之处仅仅在于对顶点访问的次序不同。

参考答案

（59）D

试题（60）

单向链表中往往含有一个头结点，该结点不存储数据元素，一般令链表的头指针指向该结点，而该结点指针域的值为第一个元素结点的指针。以下关于单链表头结点的叙述中，错误的是__（60）__。

（60）A. 若在头结点中存入链表长度值，则求链表长度运算的时间复杂度为 $O(1)$

B. 在链表的任何一个元素前后进行插入和删除操作可用一致的方式进行处理

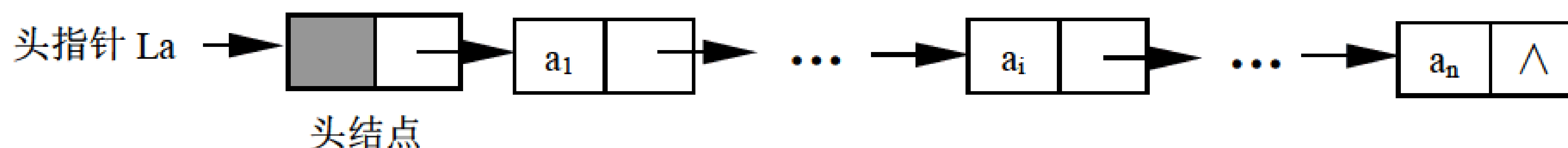
C. 加入头结点后，代表链表的头指针不因为链表为空而改变

D. 加入头结点后，在链表中进行查找运算的时间复杂度为 $O(1)$

试题（60）分析

本题考查数据结构基础知识。

含有头结点的单链表如下图所示。



在链表中加入头结点后，查找表中某一元素仍然要从头指针出发，顺序找到目标元素或失败时找到表尾为止，时间复杂度与表长成正比。

参考答案

（60）D

试题（61）

对于长度为 m（m>1）的指定序列，通过初始为空的一个栈、一个队列后，错误的叙述是__（61）__。

（61）A. 若入栈和入队的序列相同，则出栈序列和出队序列可能相同

B. 若入栈和入队的序列相同，则出栈序列和出队序列可以互为逆序

C. 入队序列与出队序列关系为 1:1，而入栈序列与出栈序列关系是 1:n(n≥1)

D. 入栈序列与出栈序列关系为 1:1，而入队序列与出队序列关系是 1:n(n≥1)

试题（61）分析

本题考查数据结构基础知识。

队列的元素按特点是先进先出。对于队列，元素的进入次序和出队的次序相同，例如，入队的序列为 a、b、c，则出队的序列也为 a、b、c。对于栈则不同，栈的运算特点是后进先出。若入栈序列为 a、b、c，则出栈序列可能为 a、b、c，a、c、b，b、a、c，b、c、a 或者 c、b、a，而 c、a、b 则不行，因此，入栈序列与出栈序列关系为 1:1，而入队序列与出队序列关系为 1:n(n≥1)。

参考答案

（61）D

试题（62）

字符串采用链表存储方式时，每个结点存储多个字符有助于提高存储密度。若采用结点大小相同的链表存储串，在串比较、求子串、串连接、串替换等串的基本运算中，__（62）__。

（62）A. 进行串的比较运算最不方便　　B. 进行求子串运算最不方便
　　C. 进行串连接最不方便　　D. 进行串替换最不方便

试题（62）分析

本题考查数据结构基础知识。

在串比较、求子串、串连接、串替换运算中，除了串替换外，其他运算都不会改变串中的内容，因此，在链表存储方式下进行串替换最不方便。

参考答案

（62）D

试题（63）

某算法的时间复杂度表达式为 $T(n)=an^2+bn\lg n+cn+d$，其中，n 为问题的规模，a、b、c 和 d 为常数，用 O 表示其渐近时间复杂度为__（63）__。

（63）A. $O(n^2)$　　B. $O(n)$　　C. $O(n\lg n)$　　D. $O(1)$

试题（63）分析

本题考查算法分析的基础知识。

在算法分析中，符号 O 用于表示算法运行时间的上限。从定义上说，对一个函数 $g(n)$，$O(g(n))$表示函数集合：

$\{f(n)$：存在正常数 c 和 $n0$，使得对所有的 $n\geqslant n0$，有 $0\leqslant f(n)\leqslant cg(n)\}$

根据上述定义，可以知道表达式 $T(n)=an^2+bn\lg n+cn+d$ 在函数集合 $O(n^2)$ 中。对此问题，简单的做法是忽略 n 的低阶项和最高阶项 n^2 的常系数，故答案应为 $O(n^2)$。

参考答案

（63）A

试题（64）、（65）

以下关于快速排序算法的描述中，错误的是__（64）__。在快速排序过程中，需要设立基准元素并划分序列来进行排序。若序列由元素{12,25,30,45,52,67,85}构成，则初始排列为__（65）__时，排序效率最高（令序列的第一个元素为基准元素）。

（64）A. 快速排序算法是不稳定的排序算法
　　B. 快速排序算法在最坏情况下的时间复杂度为 $O(n\lg n)$
　　C. 快速排序算法是一种分治算法
　　D. 当输入数据基本有序时，快速排序算法具有最坏情况下的时间复杂度

（65）A. 45,12,30,25,67,52,85　　B. 85,67,52,45,30,25,12
　　C. 12,25,30,45,52,67,85　　D. 45,12,25,30,85,67,52

试题（64）、（65）分析

本题考查快速排序算法。

快速排序算法是一种经典的排序算法，其基本思想是选择一个基准元素（通常选择第一个元素或者最后一个元素），通过一趟排序将待排序序列分成两部分，一部分比基准元素小，一部分大于等于基准元素，此时基准元素在其排好序后的正确位置；然后再递归地排序划分的两部分，因此本质上快速排序是一种分治算法。由于在排序的过程中，各元素与基准元素比较大小，若小于基准元素则与基准元素交换位置，因此该算法是不稳定的排序算法。当每一趟排序进行后，选择的基准元素恰好最大或者最小时，就把序列分成极端不均衡的两部分，即一部分为空，另一部分为待排序序列的元素个数减 1，此时算法处于最坏情况，其时间复杂度为 $O(n^2)$。当输入数据基本有序或者所有元素值相等时，不论选择第一个元素还是最后一个元素作为基准元素，都恰好把序列分成极端不均衡的两部分，快速排序算法具有最坏情况下的时间复杂度。

对于选项 A，以 45 作为基准元素进行第一趟划分，先从后向前找出比 45 小的元素，67、52、85 这三个元素保持不动，找到 25，将其与 45 交换后，第一趟划分完成，序列为 25,12,30,45,67,52,85。第二趟先对子序列 25,12,30 进行划分，使得 25 与 12 对调，形成子序列 12,25,30；然后对 67,52,85 进行划分，使得 67 与 52 交换，形成子序列 52,67,85。至此，整个排序过程完成。期间，第一趟划分中元素的比较次数为 6 次、交换 1 次，第二趟划分中元素的比较次数共 4 次、交换次数为两次，因此，排序过程中比较次数共 10 次，交换次数为 3 次。

对于选项 B，以 85 作为基准元素，因 12 比它小，所以将 85 与 12 交换，由于剩下的元素都比 85 小，因此保持不动，第一趟划分之后的元素序列为 12,67,52,45,30,25,85，期间元素比较次数为 6 次、交换 1 次，第二趟对 85 之前的 6 个元素进行划分，由于 67、52、45、30、25 都比基准元素 12 大，因此它们保持不动，完成第二趟划分，形成的子序列为 12,67,52,45,30,25，期间比较次数为 5、交换次数为 0。接下来第三趟对子系列 67,52,45,30,25 进行划分，以 67 为基准元素，情况与第一趟相同，进行 4 次比较、1 次交换后，形成子序列 25,52,45,30,67。第四趟对子序列 25,52,45,30 进行划分，情况与第二趟相同。依此类推，完成排序时比较次数为 21 次（6+5+4+3+2+1）。

对于选项 C，以 12 作为基准元素，因为后面的所有元素都比它大，所以所有元素保持不动，第一趟划分之后的元素序列为 12,25,30,45,52,67,85，期间元素比较次数为 6 次、交换 0 次。第二趟对子序列 25,30,45,52,67,85 进行划分，以 25 作为基准元素，因为后面的所有元素都比它大，所以所有元素保持不动，第一趟划分之后的元素序列为 25,30,45,52,67,85，期间元素比较次数为 5 次、交换 0 次。接下来对子序列 30,45,52,67,85 进行划分，同理，元素保持不动，期间元素比较次数为 4 次、交换 0 次。依此类推，完成整个排序比较次数为 21 次、交换 0 次。

对于选项 D，以 45 作为基准元素进行第一趟划分，先从后向前找出比 45 小的元素，

85、67、52 这三个元素保持不动，找到 30，将其与 45 交换后，第一趟划分完成，序列为 30,12,25,45,85,67,52，期间元素比较次数为 6 次、交换 1 次。第二趟先对子序列 30,12,25 进行划分，以 30 为基准元素，30 与 25 交换，经过 2 次比较、1 次交换后子序列为 25,12,30，需要再次对子序列 25,12 进行划分；同理，对子序列 85,67,52 进行划分后，结果为 51,67,87，还需对子序列 51,67 进行划分。排序过程中比较次数共 12 次。

参考答案

（64）B　（65）A

试题（66）、（67）

下列网络互连设备中，属于物理层的是__（66）__，属于网络层的是__（67）__。

（66）A. 中继器　B. 交换机　C. 路由器　D. 网桥

（67）A. 中继器　B. 交换机　C. 路由器　D. 网桥

试题（66）、（67）分析

中继器是网络层设备，其作用是对接收的信号进行再生放大，以延长传输的距离。网桥是数据链路层设备，可以识别 MAC 地址，进行帧转发。交换机是由硬件构成的多端口网桥，也是一种数据链路层设备。路由器是网络层设备，可以识别 IP 地址，进行数据包的转发。

参考答案

（66）A　（67）C

试题（68）～（70）

下图是 HTML 文件 test.html 在 IE 中的显示效果，实现图中①处效果的 HTML 语句是__（68）__，实现图中②处效果的 HTML 语句是__（69）__，实现图中③处效果的 HTML 语句是__（70）__。



（68）A．<TITLE>我的主页</TITLE>　　B．<HEAD>我的主页</HEAD >
　　C．<BODY>我的主页</BODY >　　D．<H1>我的主页</H1>
（69）A．<HR>　　B．<LINE> </LINE >
　　C．<CELL> </CELL >　　D．<TR> </TR >
（70）A．<B>Welcome</B>　　B．<UL>Welcome</UL>
　　C．<I>Welcome</I>　　D．<H>Welcome</H>

试题（68）～（70）分析

本题考查 HTML 标记相关知识。

①处显示的信息为文档的标题，采用<TITLE> </TITLE>标记。②处为一条分隔线，采用<HR>标记。③处字体 Welcome 是斜体，采用<I></I>标记。

参考答案

（68）A　（69）A　（70）C

试题（71）～（75）

Why is （71） fun? What delights may its practitioner expect as his reward? First is the sheer joy of making things. As the child delights in his mud pie, so the adult enjoys building things, especially things of his own design. Second is the pleasure of making things that are useful to other people. Third is the fascination of fashioning complex puzzle-like objects of interlocking moving parts and watching them work in subtle cycles, playing out the consequences of principles built in from the beginning. Fourth is the joy of always learning, which springs from the （72） nature of the task. In one way or another the problem is ever new, and its solver learns something: sometimes （73）, sometimes theoretical, and sometimes both. Finally, there is the delight of working in such a tractable medium. The （74）, like the poet, works only slightly removed from pure thought-stuff. Few media of creation are so flexible, so easy to polish and rework, so readily capable of realizing grand conceptual structures.

Yet the program （75）, unlike the poet's words, is real in the sense that it moves and works, producing visible outputs separate from the construct itself. It prints results, draws pictures, produces sounds, moves arms. Programming then is fun because it gratifies creative longings built deep within us and delights sensibilities we have in common with all men.

（71）A．programming　B．composing　C．working　D．writing
（72）A．repeating　B．basic　C．non-repeating　D．advance
（73）A．semantic　B．practical　C．lexical　D．syntactical
（74）A．poet　B．architect　C．doctor　D．programmer
（75）A．construct　B．code　C．size　D．scale

参考译文

编程为什么有趣？作为回报，其从业者期望得到什么样的快乐？首先是一种创建事物的纯粹快乐。如同小孩在玩泥巴时感到愉快一样，成年人喜欢创建事物，特别是自己进行设计。其次，快乐来自于开发对其他人有用的东西。第三是整个过程体现出魔术般的力量——将相互啮合的零部件组装在一起，看到它们精妙地运行，得到预先所希望的结果。第四是学习的乐趣，来自于这项工作的非重复特性。人们所面临的问题，在某个或其他方面总有些不同，因而解决问题的人可以从中学习新的事物：有时是实践上的，有时是理论上的，或者兼而有之。最后，乐趣还来自于工作在如此易于驾驭的介质上。程序员就像诗人一样，几乎仅仅工作在单纯的思考中，凭空地运用自己的想象来建造自己的“城堡”。很少有这样的介质——创造的方式如此灵活，如此易于精炼和重建，如此容易地实现概念上的设想。

然而程序毕竟同诗歌不同，它是实实在在的东西；可以移动和运行，能独立产生可见的输出；能打印结果，绘制图形，发出声音，移动支架。编程非常有趣，在于它不仅满足了我们内心深处进行创造的渴望，而且还愉悦了每个人内在的情感。

参考答案

（71）A （72）C （73）B （74）D （75）A

第 4 章　2009 下半年软件设计师下午试题分析与解答

试题一（共 15 分）

阅读以下说明和数据流图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

【说明】

现准备为某银行开发一个信用卡管理系统 CCMS，该系统的基本功能为：

1．信用卡申请。非信用卡客户填写信用卡申请表，说明所要申请的信用卡类型及申请者的基本信息，提交 CCMS。如果信用卡申请被银行接受，CCMS 将记录该客户的基本信息，并发送确认函给该客户，告知客户信用卡的有效期及信贷限额；否则该客户将会收到一封拒绝函。非信用卡客户收到确认函后成为信用卡客户。

2．信用卡激活。信用卡客户向 CCMS 提交激活请求，用信用卡号和密码激活该信用卡。激活操作结束后，CCMS 将激活通知发送给客户，告知客户其信用卡是否被成功激活。

3．信用卡客户信息管理。信用卡客户的个人信息可以在 CCMS 中进行在线管理。每位信用卡客户可以在线查询和修改个人信息。

4．交易信息查询。信用卡客户使用信用卡进行的每一笔交易都会记录在 CCMS 中。信用卡客户可以通过 CCMS 查询并核实其交易信息（包括信用卡交易记录及交易额）。

下图（a）和（b）分别给出了该系统的顶层数据流图和 0 层数据流图的初稿。

【问题 1】

根据说明，将图（a）中的 E1～E3 填充完整。

【问题 2】

图（a）中缺少三条数据流，根据说明，分别指出这三条数据流的起点和终点。（注：数据流的起点和终点均采用图中的符号和描述）

【问题 3】

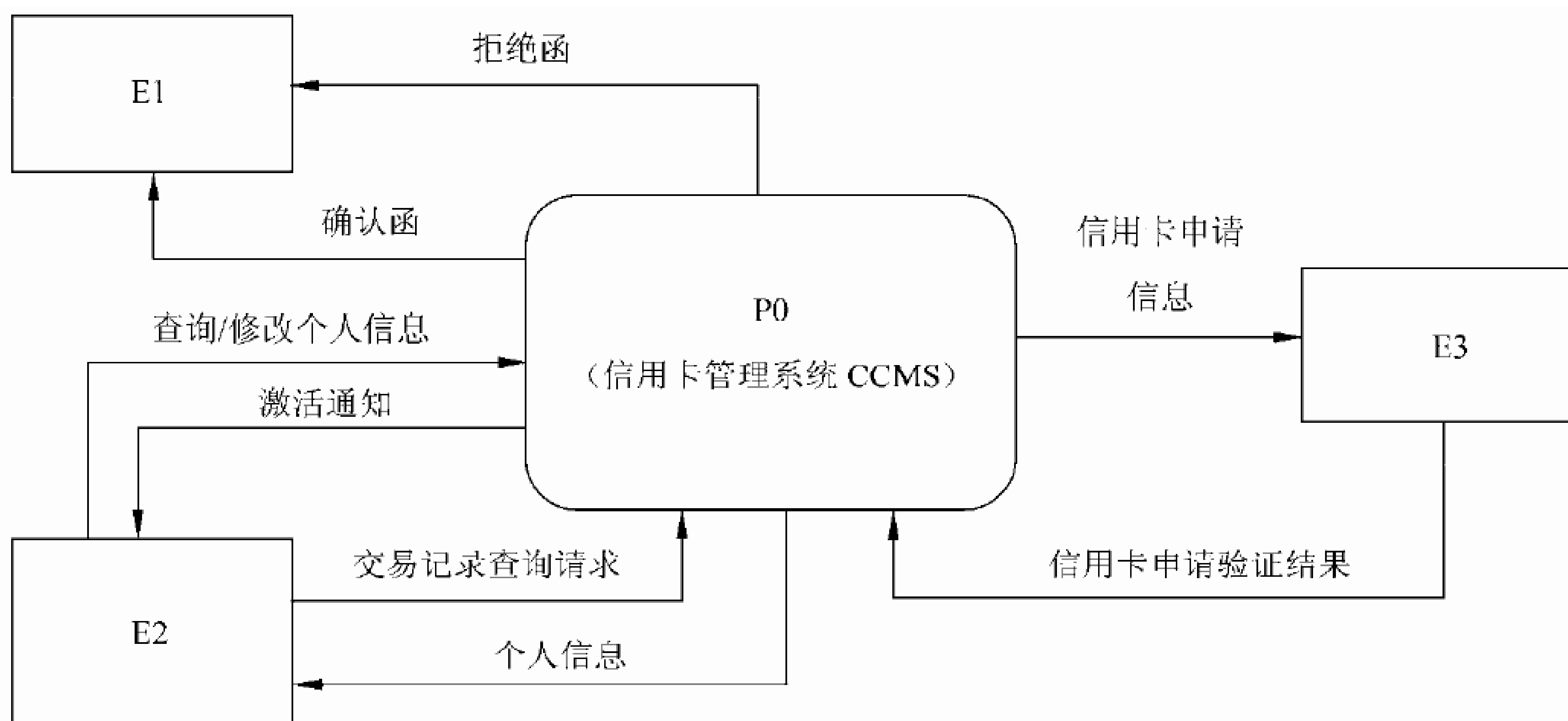
图（b）中有两条数据流是错误的，请指出这两条数据流的名称，并改正。（注：数据流的起点和终点均采用图中的符号和描述）

【问题 4】

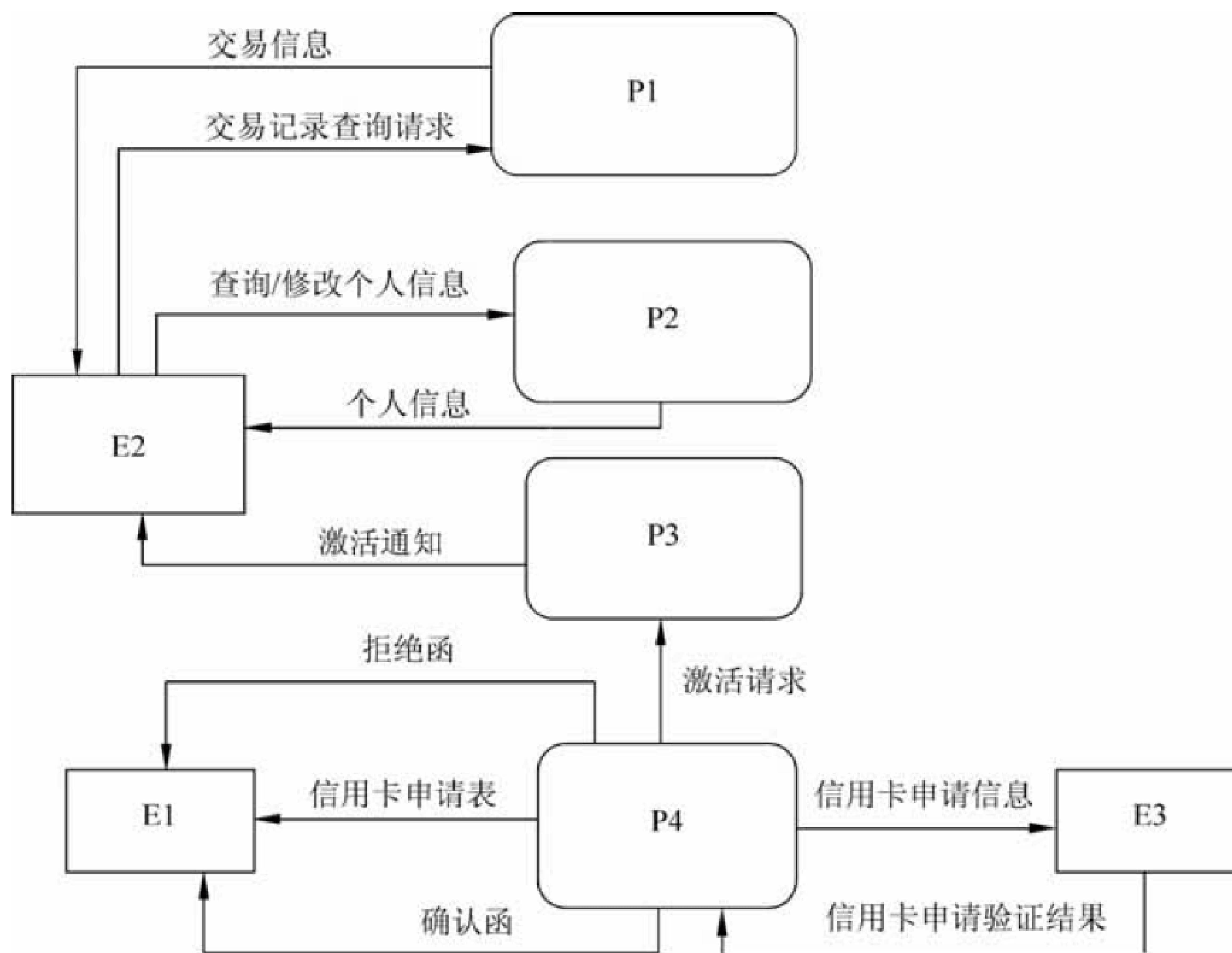
根据说明，将图（b）中 P1～P4 的处理名称填充完整。

试题一分析

本题属于经典的考题，主要考查对 DFD 的理解。



（a）顶层数据流图



（b）0 层数据流图

【问题 1】

根据题目中的说明，可以很容易找到与 CCMS 系统进行信息交互的角色有非信用卡客户、信用卡客户以及银行。下面要做的事情是在上图（a）中找到对应的位置。

根据图（a）给出的输入和输出数据流，可知 E1 表示非信用卡客户；E2 表示信用卡客户；E3 表示银行。

【问题 2】

这道题目主要考查父图与子图的平衡问题。对照上图（a）和（b）可以发现，数据流“信用卡申请表”“激活请求”“信用卡交易信息”出现在图（b）中，却没有出现在图（a）中。下一步只要正确地标出这三条数据流的起点和终点就可以了。

【问题 3】

数据流的错误主要有与错误的加工相连接、没有经过任何的加工、数据流方向错误等。在图（b）中，并没有出现任何的数据流没有经过加工，那错误就在于与数据流相连接的加工有问题或者数据流方向错误。

这样，可以找两条有错误的数据流“激活请求”和“信用卡申请表”。从图（a）中可知，“激活请求”是从系统流向外部实体 E2 的，而在图（b）中，“激活请求”却出现在两个加工之间。数据流“信用卡申请表”是在问题 2 中补充找到的数据流，它应该从外部实体 E1 流向 CCMS 系统。

【问题 4】

这道题要求将图（b）中的加工补充完整。加工的名称在说明中已经明确给出了：信用卡申请、信用卡激活、信用卡客户信息管理以及交易信息查询。下一步需要根据图（b）中给出的数据流关系将这 4 个加工对号入座即可。这样可以得到 P1 表示交易信息查询；P2 表示信用卡客户信息管理；P3 表示信用卡激活；P4 表示信用卡申请。

参考答案

【问题 1】

E1：非信用卡客户

E2：信用卡客户

E3：银行

【问题 2】

起　点	终　点	名　称
E1	P0	信用卡申请表
E2	P0	激活请求
P0 或 信用卡管理系统 CCMS	E2	信用卡交易信息

注：每条数据流的起点和终点全部答对方可给 1 分

【问题 3】

错误的数据流：

错误数据流名称是激活请求和信用卡申请表。

改正后的数据流：

数据流名称	改正后数据流起点	改正后数据流终点
激活请求	E2	P3
信用卡申请表	E1	P4

【问题 4】

P1：交易信息查询　　　P2：客户信息管理

P3：信用卡激活　　　　P4：信用卡申请

试题二（共 15 分）

阅读下列说明，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某公司拟开发一多用户电子邮件客户端系统，部分功能的初步需求分析结果如下：

（1）邮件客户端系统支持多个用户，用户信息主要包括用户名和用户密码，且系统中的用户名不可重复。

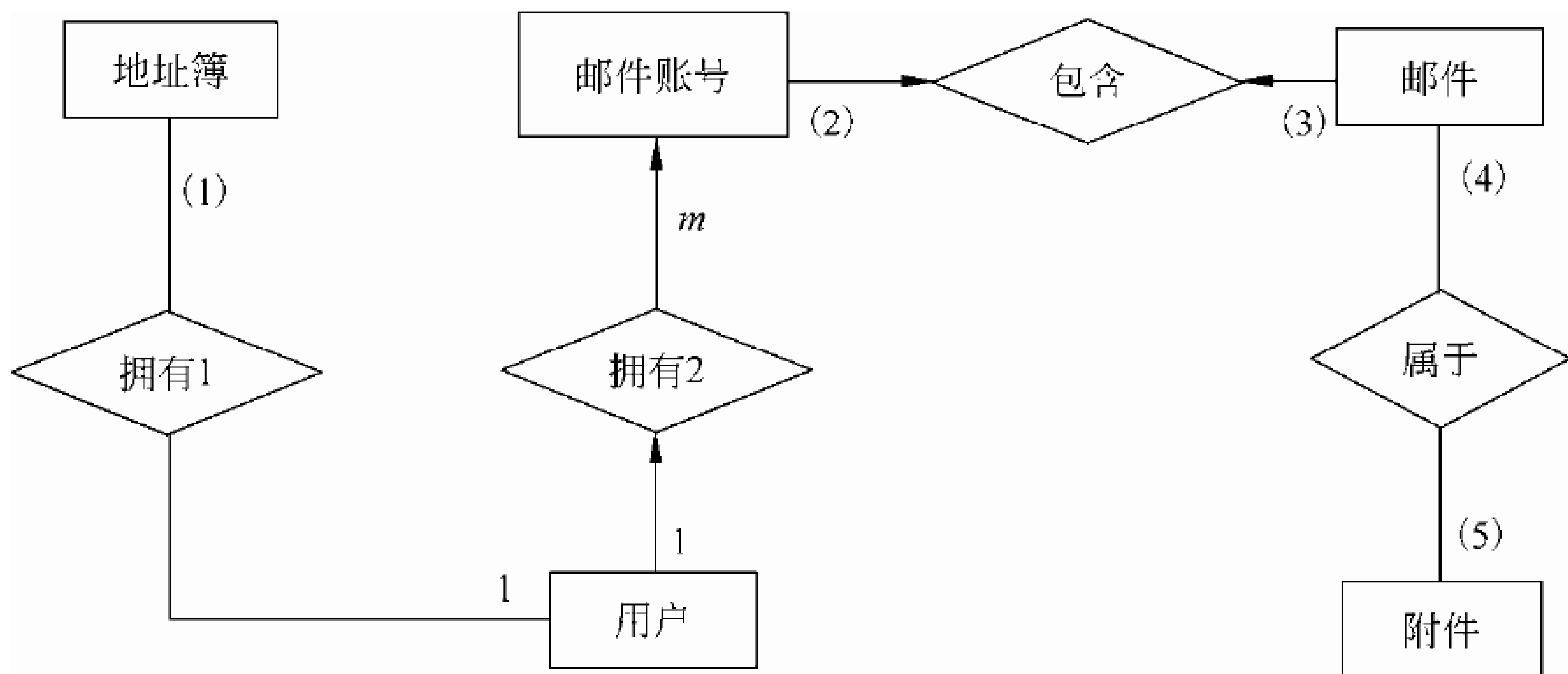
（2）邮件账号信息包括邮件地址及其相应的密码，一个用户可以拥有多个邮件地址（如 user1@123.com）。

（3）一个用户可拥有一个地址簿，地址簿信息包括联系人编号、姓名、电话、单位地址、邮件地址 1、邮件地址 2、邮件地址 3 等信息。地址簿中一个联系人只能属于一个用户，且联系人编号唯一标识一个联系人。

（4）一个邮件账号可以含有多封邮件，一封邮件可以含有多个附件。邮件主要包括邮件号、发件人地址、收件人地址、邮件状态、邮件主题、邮件内容、发送时间、接收时间。其中，邮件号在整个系统内唯一标识一封邮件，邮件状态有已接收、待发送、已发送和已删除 4 种，分别表示邮件是属于收件箱、发件箱、已发送箱和废件箱。一封邮件可以发送给多个用户。附件信息主要包括附件号、附件文件名、附件大小。一个附件只属于一封邮件，附件号仅在一封邮件内唯一。

【问题 1】

根据以上说明设计的 E-R 图如下图所示，请指出地址簿与用户、电子邮件账号与邮件、邮件与附件之间的联系类型。



电子邮件客户端系统 E-R 图

【问题 2】

该邮件客户端系统的主要关系模式如下，请填补（a）～（c）的空缺部分。

用户（用户名，用户密码）
地址簿（<u>（a）</u>，联系人编号，姓名，电话，单位地址，邮件地址 1，邮件地址 2，邮件地址 3）
邮件账号（邮件地址，邮件密码，用户名）
邮件（<u>（b）</u>，收件人地址，邮件状态，邮件主题，邮件内容，发送时间，接收时间）
附件（<u>（c）</u>，附件号，附件文件名，附件大小）

【问题 3】

（1）请指出问题 2 中给出的地址簿、邮件和附件关系模式的主键，如果关系模式存在外键请指出。

（2）附件属于弱实体吗？请用 50 字以内的文字说明原因。

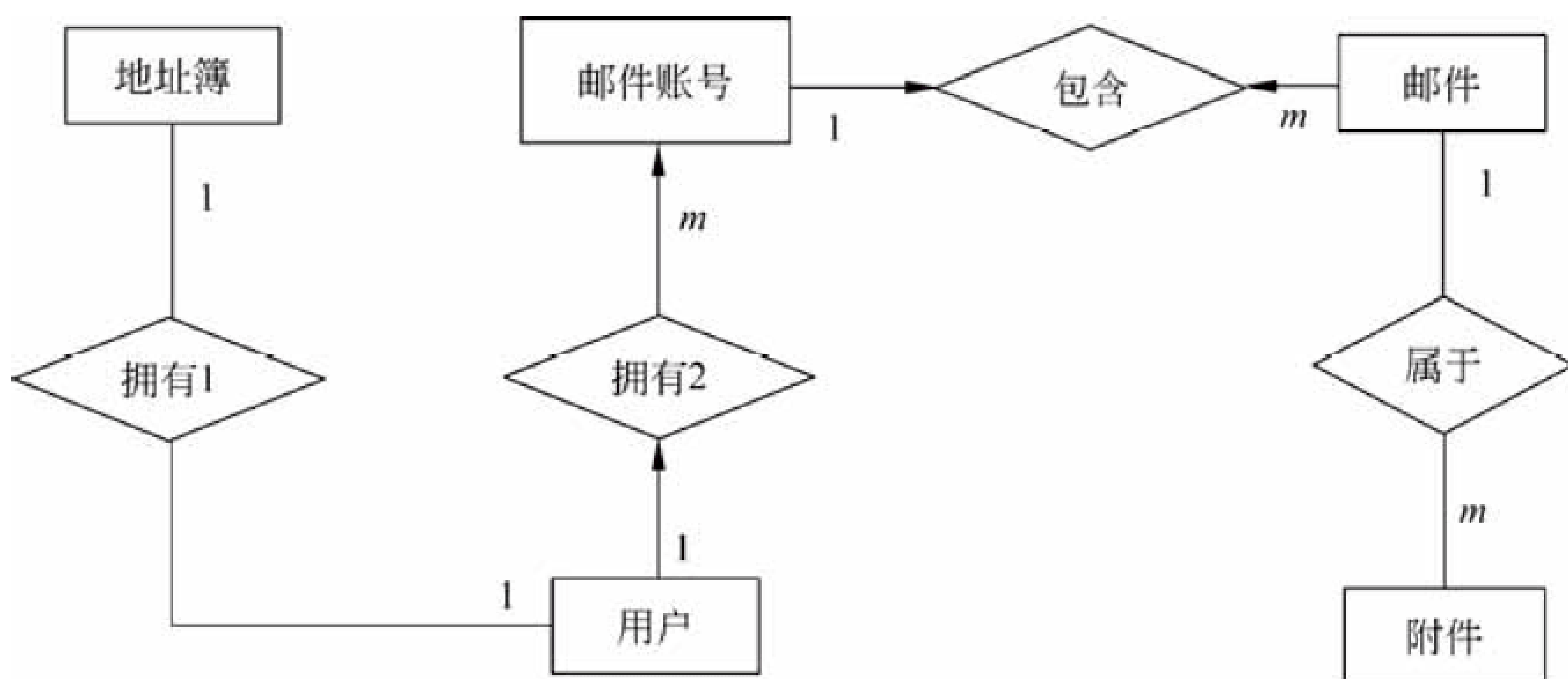
试题二分析

本题考查数据库系统中实体联系模型（E-R 模型）的设计和关系模式的设计。

【问题 1】

两个实体模型之间的联系可以分为三类：一对一联系（1∶1）、一对多联系（1∶n）和多对多联系（m∶n）。

根据题意，地址簿与用户之间应该是一个 1∶1 的联系，空（1）应填 1。电子邮件账号与邮件之间应该是一个 1∶m 的联系，故空（2）和空（3）应分别填写 1 和 m。邮件与附件之间应该是一个 1∶m 的联系，故空（4）和空（5）应分别填写 1 和 m。得到的 E-R 图如下图所示。



【问题 2】

空（a）分析：根据题意可知邮件客户端系统支持多个用户，用户信息主要包括用户名和用户密码，且系统中的用户名不可重复，“用户名”可以作为用户关系模式主键。地

址簿关系模式中与用户关系模式是一个1∶1的联系，必须将任一方的主键加入另一方，以建立它们之间的联系，故空（a）处应填写“用户名”。

空（b）分析：根据题意可知邮件号在整个系统内唯一标识一封邮件，故邮件关系模式必须有属性“邮件号”，另外一封邮件需要填写“发件人地址”，故空（b）处应填写“邮件号，发件人地址”。

空（c）分析：根据题意可知邮件和附件是一个1∶*m*的联系，按照E-R模型向关系模型的转换规则对于1∶*m*的联系应将1端的主键并入多端，故空（c）处应填写“邮件号”。

【问题3】

（1）地址簿关系模式的主键为“联系人编号”，外键为“用户名”，因为“用户名”是参考用户关系模式的“用户名”主键。邮件关系模式的主键为“邮件号”，外键为“发件人地址”或“收件人地址”，因为当用户向其他人发邮件的时候，“发件人地址”是参考邮件账号关系模式的“邮件地址”的主键；当用户收邮件的时候，“收件人地址”是参考邮件账号关系模式的“邮件地址”的主键。附件关系模式的主键为“邮件号，附件号”，外键为“邮件号”，因为该“邮件号”参考邮件关系模式的“邮件号”的主键。

（2）附件属于弱实体，因为如果没有邮件，附件也就不存在。

参考答案

【问题1】

（1）1

（2）1

（3）*m*　或*n*或 *

（4）1

（5）*m*　或*n*或 *

【问题2】

（a）用户名

（b）邮件号，发件人地址

（c）邮件号

【问题3】

（1）

关系模式	主　键	外　键
地址簿	联系人编号	用户名
邮件	邮件号	发件人地址或收件人地址
附件	邮件号，附件号	邮件号

（2）附件属于弱实体，因为附件的存在必须以邮件的存在为前提，即附件总是依附于某邮件。

试题三（共 15 分）

阅读下列说明和 UML 图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

【说明】

某企业为了方便员工用餐，为餐厅开发了一个订餐系统（COS：Cafeteria Ordering System），企业员工可通过企业内联网使用该系统。

企业的任何员工都可以查看菜单和今日特价。

系统的顾客是注册到系统的员工，可以订餐（如果未登录，需先登录）、注册工资支付、预约规律的订餐，在特殊情况下可以覆盖预订。

餐厅员工是特殊顾客，可以进行备餐、生成付费请求和请求送餐，其中对于注册工资支付的顾客生成付费请求并发送给工资系统。

菜单管理员是餐厅特定员工，可以管理菜单。

送餐员可以打印送餐说明，记录送餐信息（如送餐时间）以及记录收费（对于没有注册工资支付的顾客，由送餐员收取现金后记录）。

顾客订餐过程如下：

1. 顾客请求查看菜单；
2. 系统显示菜单和今日特价；
3. 顾客选菜；
4. 系统显示订单和价格；
5. 顾客确认订单；
6. 系统显示可送餐时间；
7. 顾客指定送餐时间、地点和支付方式；
8. 系统确认接受订单，然后发送 E-mail 给顾客以确认订餐，同时发送相关订餐信息通知给餐厅员工。

系统采用面向对象方法开发，使用 UML 进行建模。系统的顶层用例图和一次订餐的活动图初稿分别如下图（a）和（b）所示。

【问题 1】

根据说明中的描述，给出图（a）中 A1 和 A2 所对应的参与者。

【问题 2】

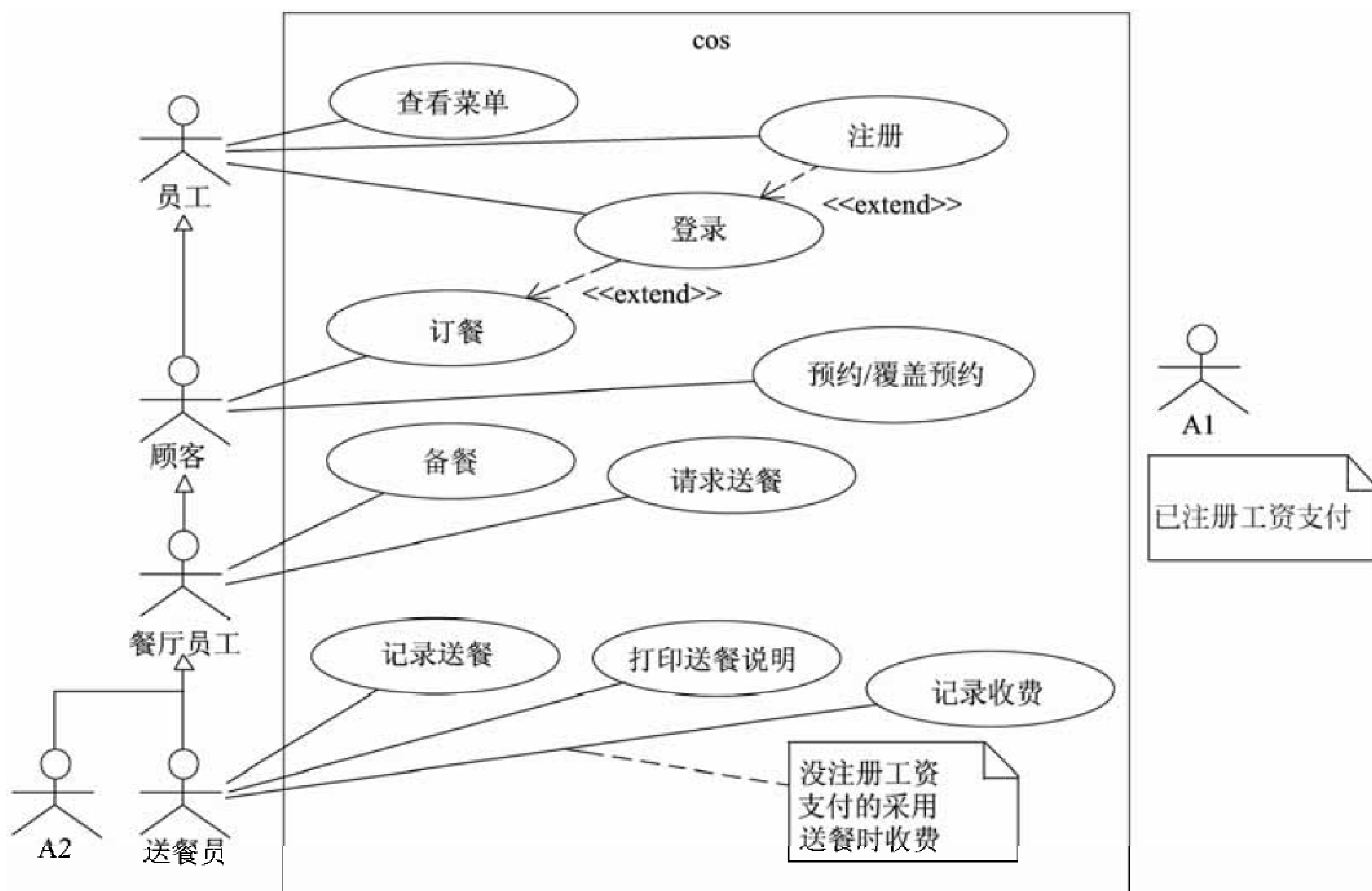
根据说明中的描述，给出图（a）中缺少的四个用例及其所对应的参与者。

【问题 3】

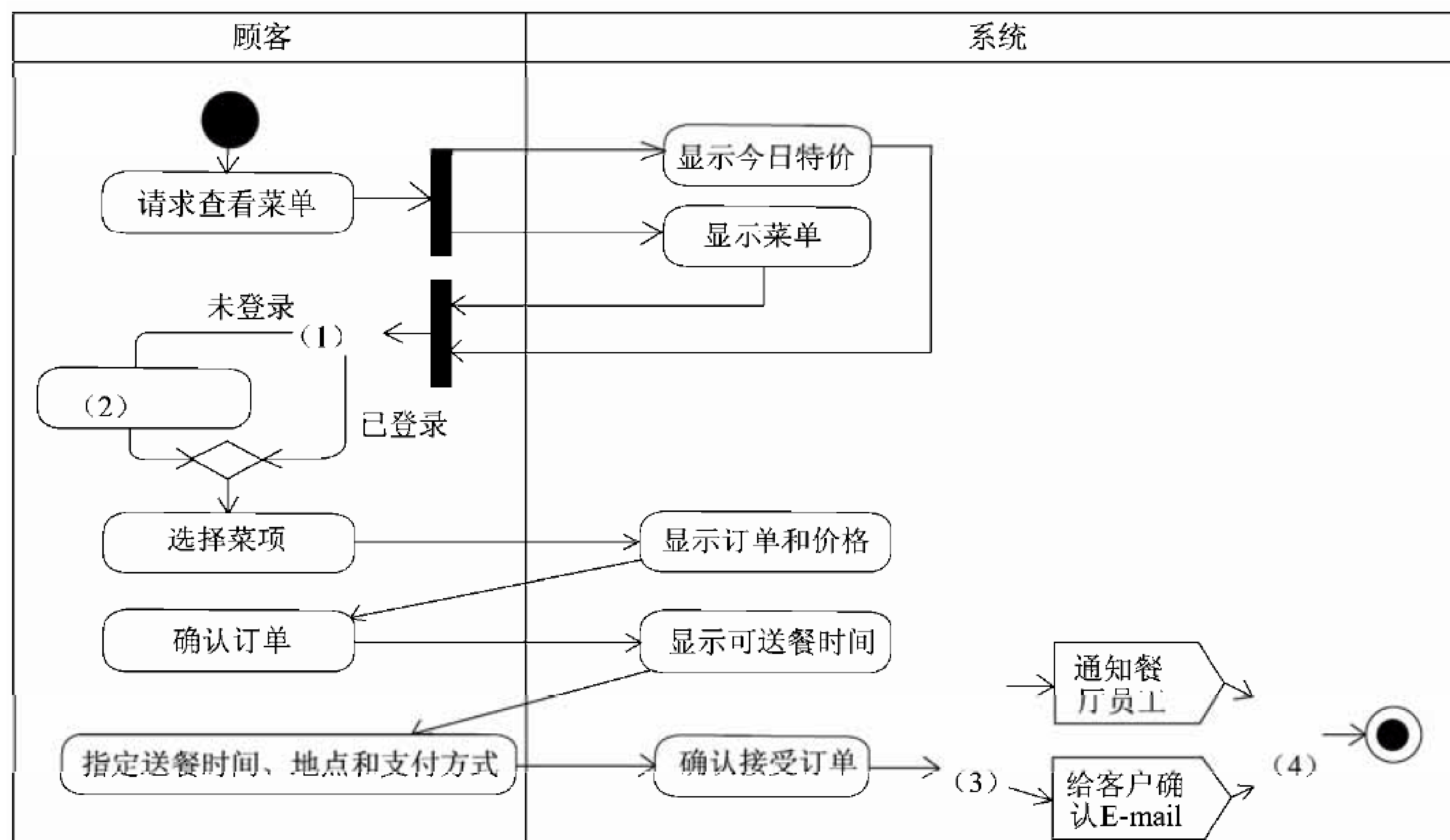
根据说明中的描述，给出图（b）中（1）～（4）处对应的活动名称或图形符号。

【问题 4】

指出图（a）中员工和顾客之间是什么关系，并解释该关系的内涵。



（a）COS 系统顶层用例图



（b）一次订餐的活动图

试题三分析

本题考查面向对象系统开发时，采用 UML 模型进行建模的方法。

此类题目要求考生认真阅读题目说明中对现实问题的描述，使用 UML 建模时的原则，从中确定用例图、活动图以及图中的各种关系。题目给出了未完成的用例图和活动图，需要根据描述给出参与者、用例、活动图中的活动和符号，以及参与者之间的关系内涵。

用例图是用例建模的一个重要产物，它以图形化的方式将系统描述成用例、参与者及其之间的关系。用例图在高层交流了系统必须处理的业务事件的范围，是描述系统与其他外部系统以及用户之间交互的图形。发起或者触发用例的外部用户称为参与者。为了完成某些业务任务，参与者发起系统活动，即用例。在构建用例图时，常用的方式是先识别参与者，然后确定用例以及用例之间的关系。

UML 活动图用于建模系统的过程步骤或活动。构造活动图通常先为用例添加开始和结束点，为用例的主要步骤添加一个活动，从每个活动到其他活动、决策点和终点添加转换，并行活动的地方添加同步条。

【问题 1】

识别参与者时，考查和系统交互的人员和外部系统。本题中，与系统交互的人员包括员工、注册到系统的员工（顾客）、餐厅员工、菜单管理员、送餐员以及工资系统。

由“菜单管理员是餐厅特定员工”以及图中 A2 和图中餐厅员工之间的“是一种”关系可知，A2 为菜单管理员；图中还缺少描述中与工资系统的交互，由“……并发送给工资系统”可知，A1 为工资系统。

【问题 2】

考查用例及其和参与者之间的关系时，通过判断哪一个特定参与者发起或者触发了与系统的哪些交互，来识别用例并建立和参与者之间的关联。

本题中，由“任何员工都可以查看菜单和今日特价”可知，图中缺少用例查看今日特价，对应参与者是员工；由“系统的顾客是……，注册工资支付、……”可知，图中缺少用例注册工资支付，对应参与者是顾客和工资系统；由 “餐厅员工是……，可以进行备餐、生成付费请求……发送给工资系统”可知，图中缺少用例“生成付费请求”，对应的参与者是餐厅员工和工资系统；由“菜单管理员是餐厅特定员工，可以管理菜单”可知，图中缺少用例管理菜单，对应的参与者是菜单管理员。

需要注意的是，在注册工资支付所对应的参与者中，虽然没有明确说明要和工资系统交互，但是由“对于注册工资支付的顾客生成付费请求并发送给工资系统”可知，工资支付是由工资系统控制，所以注册也需要和工资系统交互。

【问题 3】

在顾客订餐过程的描述中，在“顾客选菜”之前，图中缺少符号和活动。由说明中顾客“可以订餐（如果未登录，需先登录）”可以判断，在系统“显示菜单和今日特价”之后“顾客选菜”之前，需要判断（判定符号◇）当前用户身份是否为顾客，如果不是，需先登录；由“……发送 E-mail 给顾客以确认订餐，同时发送相关订餐信息通知给

餐厅员工”可知，发送 E-mail 和通知餐厅员工为并行活动，需要在前后有同步条▬（或纵向）。

【问题 4】

参与者之间的关系表示子类型“是一种”父类型，即泛化关系。其中父类型通常是一个抽象泛化的参与者，可以完成子类型可完成的共同行为，每个具体的子类型继承它，可以完成父类型参与者同样的任务，并可以补充额外的角色功能。

参考答案

【问题 1】

A1：工资系统　　A2：菜单管理员

【问题 2】

用　例　名	参　与　者
查看当日特价	员工
注册工资支付	顾客和工资系统（或顾客和 A1）
生成付费请求	餐厅员工和工资系统（或餐厅员工和 A1）
管理菜单	菜单管理员（或 A2）

（注：4 行的顺序可以不同，但是每行必须对应）

【问题 3】

（1）◇　　（2）登录

（3）▬　或　▮　　（4）▬　或　▮

【问题 4】

泛化关系（一般/特殊关系、继承关系）。泛化关系描述了一个参与者可以完成另一个参与者同样的任务，并可补充额外的角色功能。

试题四（共 15 分）

阅读下列说明，回答问题 1 至问题 2，将解答填入答题纸的对应栏内。

【说明】

0-1 背包问题可以描述为：有 n 个物品，对 $i=1, 2, \cdots, n$，第 i 个物品价值为 v_i，重量为 w_i（v_i 和 w_i 为非负数），背包容量为 W（W 为非负数），选择其中一些物品装入背包，使装入背包物品的总价值最大，即 $\max\sum_{i=1}^{n} v_i x_i$，且总重量不超过背包容量，即 $\sum_{i=1}^{n} w_i x_i \leqslant W$，其中，$x_i \in \{0,1\}$，$x_i=0$ 表示第 i 个物品不放入背包，$x_i=1$ 表示第 i 个物品放入背包。

【问题 1】

用回溯法求解此 0-1 背包问题，请填充下面伪代码中（1）～（4）处空缺。

回溯法是一种系统的搜索方法。在确定解空间后，回溯法从根结点开始，按照深度优先策略遍历解空间树，搜索满足约束条件的解。对每一个当前结点，若扩展该结点已

经不满足约束条件，则不再继续扩展。为了进一步提高算法的搜索效率，往往需要设计一个限界函数，判断并剪枝那些即使扩展了也不能得到最优解的结点。现在假设已经设计了 BOUND(v,w,k,W)函数，其中 v、w、k 和 W 分别表示当前已经获得的价值、当前背包的重量、已经确定是否选择的物品数和背包的总容量。对应于搜索树中的某个结点，该函数值表示确定了部分物品是否选择之后，对剩下的物品在满足约束条件的前提下进行选择可能获得的最大价值，若该价值小于等于当前已经得到的最优解，则该结点无须再扩展。

下面给出 0-1 背包问题的回溯算法伪代码。

函数参数说明如下：

W：背包容量；n：物品个数；w：重量数组；v：价值数组；fw：获得最大价值时背包的重量；fp：背包获得的最大价值；X：问题的最优解。

变量说明如下：

cw：当前的背包重量；cp：当前获得的价值；k：当前考虑的物品编号；Y：当前已获得的部分解。

```
BKNAP( W,n,w,v,fw,fp,X )
1  cw ← cp ← 0
2  ___(1)___
3  fp ← -1
4  while true
5     while k ≤ n and cw + w[k] ≤ W do
6         ___(2)___
7         cp ← cp + v[k]
8         Y[k] ← 1
9         k ← k + 1
10    if k > n then
11            if fp < cp then
12                    fp ← cp
13                    fw ← cw
14                    k ← n
15                    X ← Y
16     else Y(k) ← 0
17     while BOUND(cp,cw,k,W) ≤ fp do
18         while k ≠ 0 and Y(k) ≠ 1 do
19             ___(3)___
20         if k = 0 then return
21         Y[k] ← 0
22         cw ← cw - w[k]
23         cp ← cp - v[k]
24     ___(4)___
```

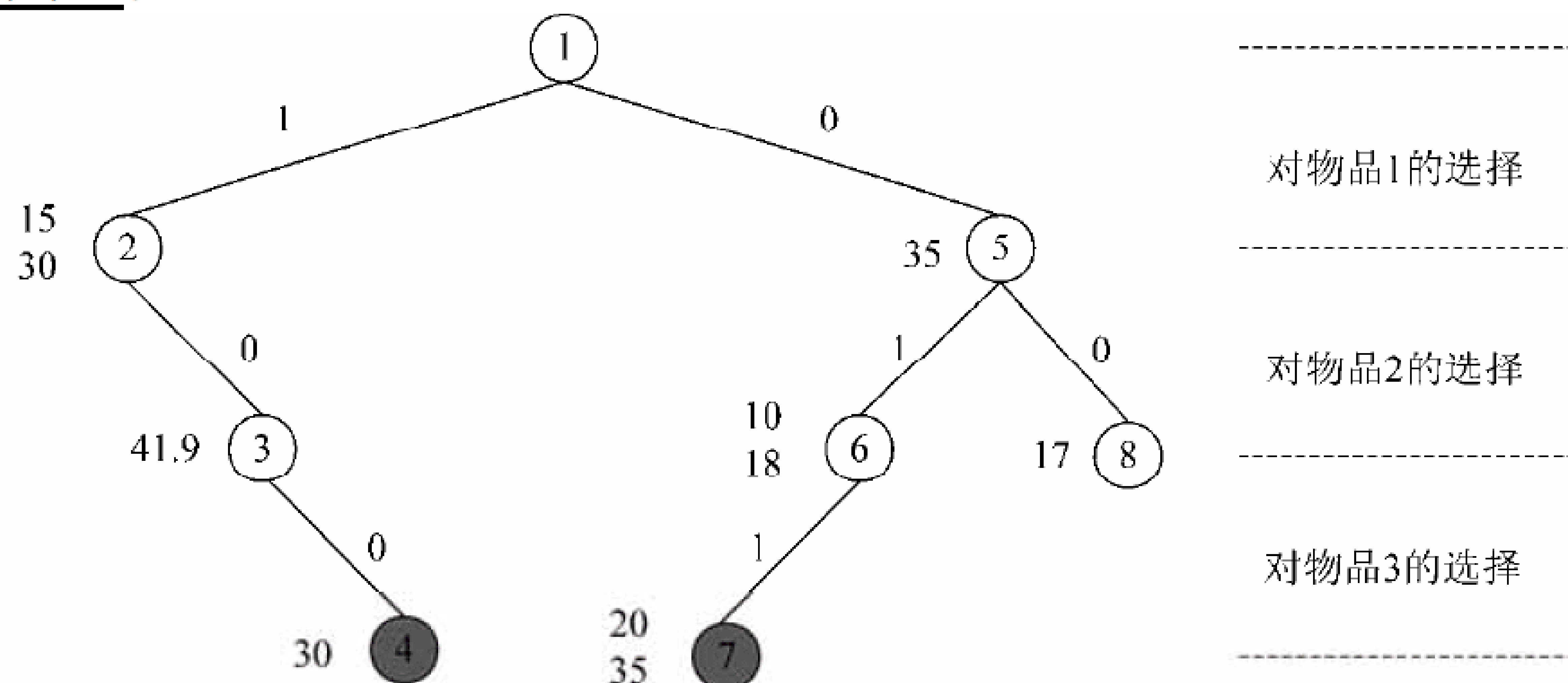
【问题 2】

考虑下表所示的实例，假设有 3 个物品，背包容量为 22。

0-1 背包问题实例

	物品 1	**物品 2**	**物品 3**
重量	15	10	10
价值	30	18	17
单位价值	2	1.8	1.7

下图是根据上述算法构造的搜索树，其中结点的编号表示了搜索树生成的顺序，边上的数字 1/0 分别表示选择/不选择对应物品。除了根结点之外，每个左孩子结点旁边的上下两个数字分别表示当前背包的重量和已获得的价值，右孩子结点旁边的数字表示扩展了该结点后最多可能获得的价值。为获得最优解，应该选择物品__（5）__，获得的价值为__（6）__。



对于上述实例，若采用穷举法搜索整个解空间，则搜索树的结点数为__（7）__，而用了上述回溯法，搜索树的结点数为__（8）__。

试题四分析

本题考查算法设计技术——回溯法。

此类题目要求考生掌握基本的算法设计技术，包括分治法、动态规划法、贪心算法、回溯法和分支限界法等，然后结合具体的问题，用对应的算法设计技术来解决问题。

【问题 1】

本题考查的是用回溯法求解 0-1 背包问题。回溯法有两类算法框架：非递归形式和递归形式，本题采用非递归形式表示。理解回溯法的基本思想和这两类算法框架是正确解答本题的根本要求。

回溯法从第一项物品开始考虑是否应该装入背包中，因此当前考虑的物品编号 k 从 1 开始，即 $k \leftarrow 1$。然后逐项往后检查，若能全部放入背包则将该项放入背包，此时背包

的重量应该是当前的重量加上当前考虑物品的重量，即 cw ← cw + w[k]，当然背包中物品的价值也为当前的价值加上当前考虑物品的价值。若已经考虑完了所有的物品，则得到一个解，判断该解是否为当前最优，若为最优，则将该解的信息放入变量 fp、fw 和 X 中。若还没有考虑完所有的物品，意味着有些物品不能放入背包，此时先判断若不将当前的物品放入背包中，则其余物品放入背包是否可能得到比当前最优解更优的解，若得不到则回溯；否则继续考虑其余的物品。

【问题 2】

根据问题 1 中给出的伪代码运行该实例，可以很容易得到此 0-1 背包问题的最优解，应该选择物品 2 和物品 3，此时背包的重量为 10+10=20，获得的价值为 17+18=35。

若采用穷举法搜索整个解空间，即要构造一棵完全二叉树，此时搜索树的结点数应为 $2^4–1=15$，而采用了上述回溯法，搜索树的结点数仅为 8 个，如上图所示。

参考答案

【问题 1】

（1）k ← 1　　或其等价形式　　（2）cw ← cw + w[k]　或其等价形式

（3）k ← k–1　　或其等价形式　　（4）k ← k+1　　或其等价形式

【问题 2】

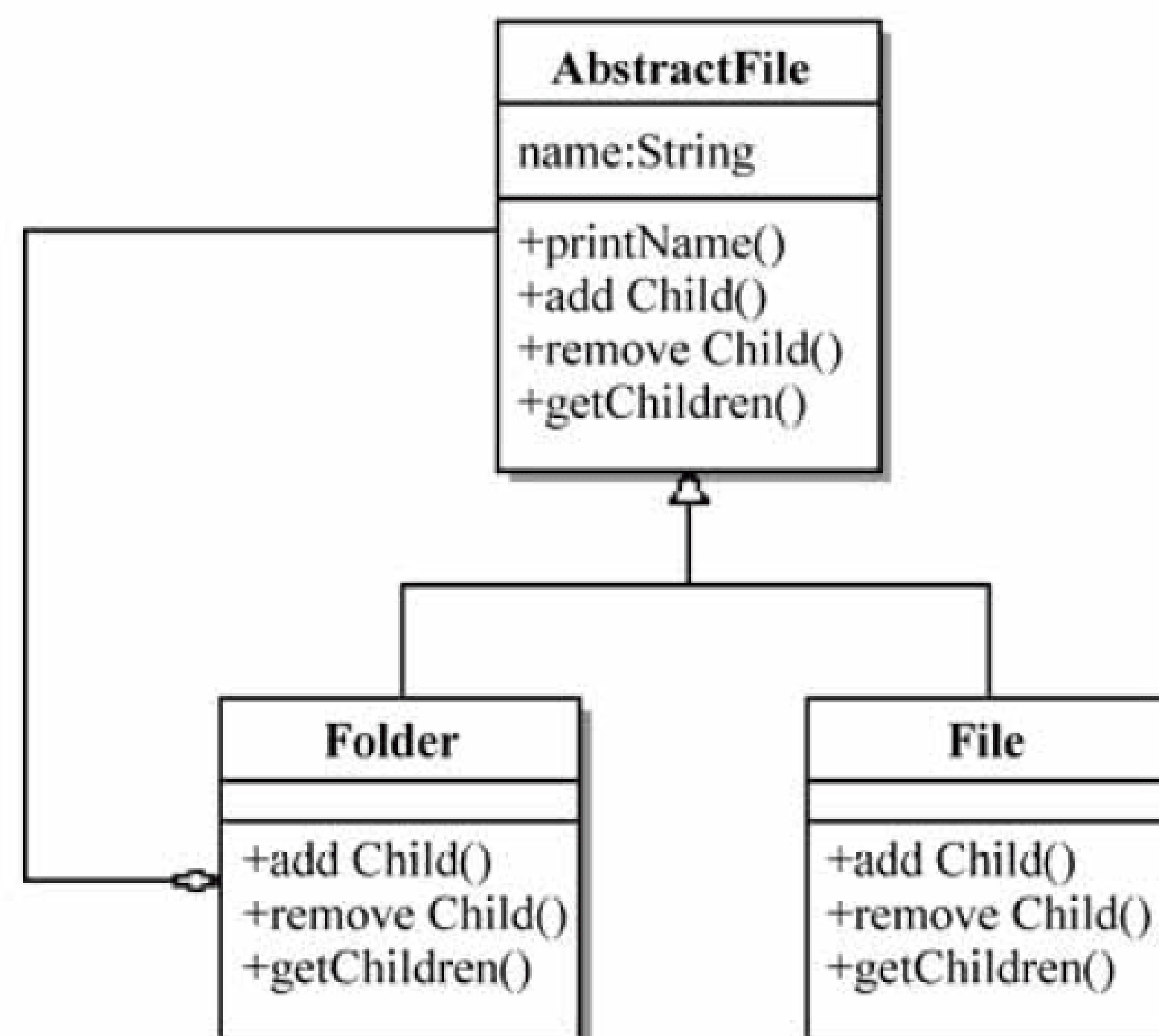
（5）物品 2 和物品 3　　（6）35　　（7）15　　（8）8

试题五（共 15 分）

阅读下列说明和 C++代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

现欲构造一文件/目录树，采用组合（Composite）设计模式来设计，得到的类图如下图所示：



【C++代码】

```
#include <list>
#include <iostream>
#include <string>
using namespace std;

class AbstractFile {
protected :
    string name;  // 文件或目录名称
public:
    void printName(){cout << name;}         // 打印文件或目录名称
    virtual void addChild(AbstractFile  *file)=0;
                                            // 给一个目录增加子目录或文件
    virtual void removeChild(AbstractFile *file)=0;
                                            // 删除一个目录的子目录或文件
    virtual list<AbstractFile*> *getChildren()=0;
                                            // 获得一个目录的子目录或文件
};

class File : public AbstractFile {
public :
    File(string name) {__(1)__ = name; }
    void addChild(AbstractFile  *file) { return ; }
    void removeChild(AbstractFile *file) { return ; }
    __(2)__ getChildren() { return __(3)__; }
};
class Folder :public AbstractFile {
private :
    list <AbstractFile*> childList;  // 存储子目录或文件
public :
    Folder(string name) {__(4)__ = name; }
    void addChild(AbstractFile *file) { childList.push_back(file); }
    void removeChild(AbstractFile *file) { childList.remove(file);}
    list<AbstractFile*> *getChildren() { return __(5)__; }
};
void main() {
```

```
        // 构造一个树形的文件/目录结构
        AbstractFile *rootFolder = new Folder("c:\\");
        AbstractFile *compositeFolder = new Folder("composite");
        AbstractFile *windowsFolder = new Folder("windows");
        AbstractFile *file = new File("TestComposite.java");
        rootFolder->addChild(compositeFolder);
        rootFolder->addChild(windowsFolder);
        compositeFolder->addChild(file);
    }
```

试题五分析

本题考查基本面向对象设计中设计模式的运用能力。

组合设计模式主要是表达整体和部分的关系，并且对整体和部分对象的使用无差别。题目中 AbstractFile 是 File 类和 Folder 类的父类，它抽象了两个类的共有属性和行为，在后续 main 方法的使用中，不论是 File 对象还是 Folder 对象，都可被当作 AbstractFile 对象来使用。另外，由于 Folder 对象可以聚合其他的 Folder 对象和 File 对象，等价于 Folder 对象可以聚合另一个 AbstractFile 对象。

在类 File 和类 Folder 的构造函数中都需要记录文件或目录的名称，因此空（1）和空（4）处主要是设置对象的名称。因为 File 对象不再聚合其他的对象，所以 File 对象没有孩子节点，因此，空（3）处应该返回 NULL。getChildren()方法继承自 AbstractFile 类，因此其返回类型也应保持一致。对于空（5），要求返回 Folder 对象的孩子对象，因此返回其成员 childList 的地址。

参考答案

（1）this->name

（2）list<AbstractFile*> *

（3）NULL

（4）this->name

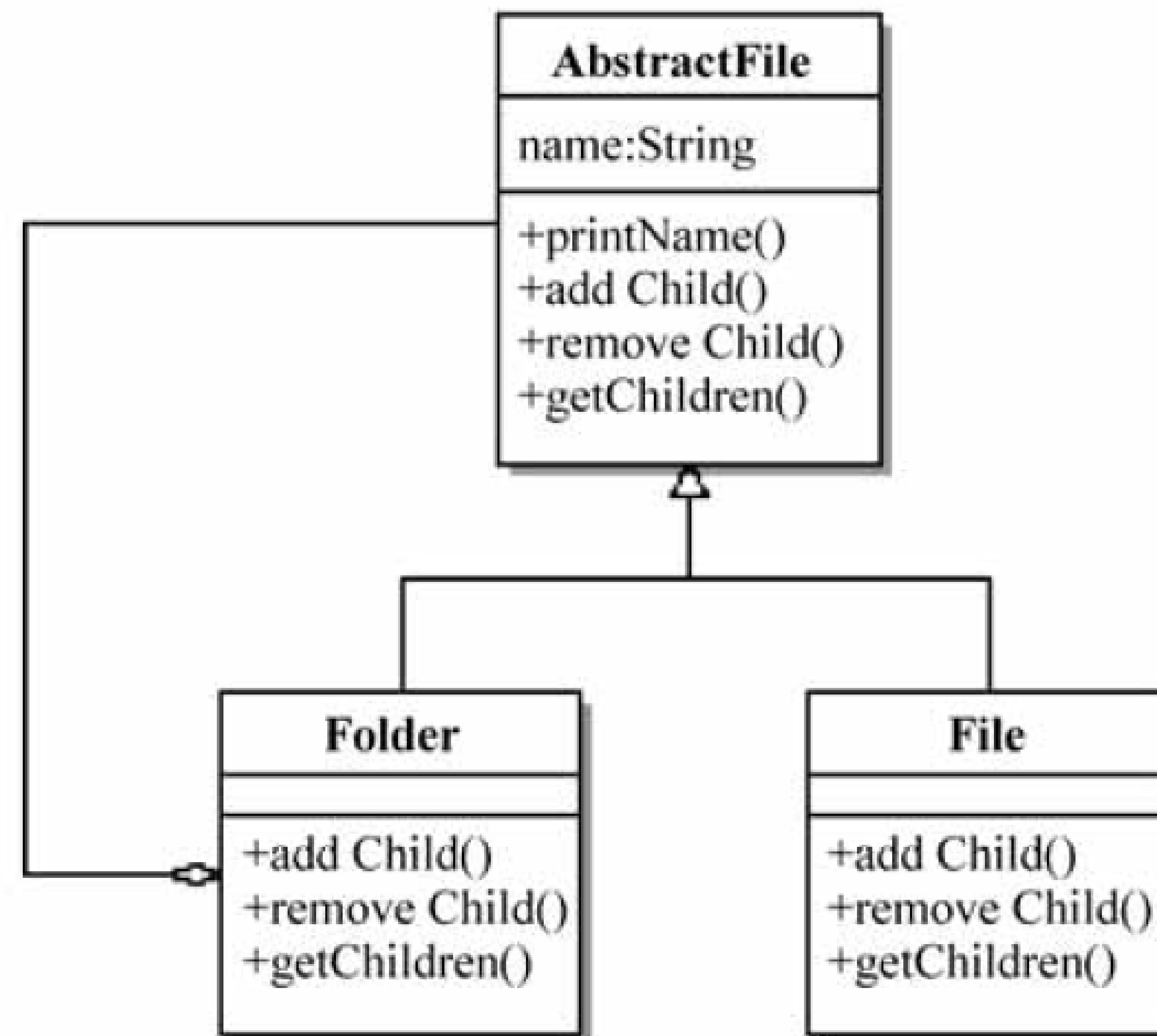
（5）&childList

试题六（共 15 分）

阅读下列说明和 Java 代码，将应填入＿（n）＿处的字句写在答题纸的对应栏内。

【说明】

现欲构造一文件/目录树，采用组合（Composite）设计模式来设计，得到的类图如下图所示：



【Java 代码】

```
import java.util.ArrayList;
import java.util.List;
__(1)__ class AbstractFile {
    protected String name;
   public void printName(){System.out.println(name);}
   public abstract boolean addChild(AbstractFile file);
   public abstract boolean removeChild(AbstractFile file);
   public abstract List<AbstractFile> getChildren();
}
class File extends AbstractFile {
   public File(String name) { this.name = name; }
   public boolean addChild(AbstractFile file) { return false; }
   public boolean removeChild(AbstractFile file) { return false; }
   public List<AbstractFile> getChildren() { return __(2)__; }
}
class Folder extends AbstractFile {
   private List <AbstractFile> childList;
   public Folder(String name) {
      this.name = name;
      this.childList = new ArrayList<AbstractFile>();
   }
   public boolean addChild(AbstractFile file) { return childList.add
   (file); }
   public boolean removeChild(AbstractFile file) { return childList.
   remove(file); }
   public __(3)__<AbstractFile> getChildren() { return __(4)__; }
```

```
}
public class Client {
    public static void main(String[] args) {
        // 构造一个树形的文件/目录结构
     AbstractFile rootFolder = new Folder("c:\\");
     AbstractFile compositeFolder = new Folder("composite");
     AbstractFile windowsFolder = new Folder("windows");
     AbstractFile file = new File("TestComposite.java");
        rootFolder.addChild(compositeFolder);
        rootFolder.addChild(windowsFolder);
        compositeFolder.addChild(file);

        // 打印目录文件树
        printTree(rootFolder);
    }
    private static void printTree(AbstractFile ifile) {
     ifile.printName();
        List <AbstractFile> children = ifile.getChildren();
        if(children == null)   return;
        for (AbstractFile file:children) {
          __(5)__;
        }
    }
}
```

该程序运行后输出结果为：

```
c:\
composite
TestComposite.java
Windows
```

试题六分析

本题考查基本面向对象设计中设计模式的运用能力。

组合设计模式主要是表达整体和部分的关系，并且对整体和部分对象的使用无差别。题目中 AbstractFile 是 File 类和 Folder 类的父类，它抽象了两个类的共有属性和行为，在后续 main 方法的使用中，不论是 File 对象还是 Folder 对象，都可被当作 AbstractFile 对象来使用。另外，由于 Folder 对象可以聚合其他的 Folder 对象和 File 对象，等价于 Folder 对象可以聚合另一个 AbstractFile 对象。

题目中 AbstractFile 类应该为抽象类，因此其修饰符应该包括 abstract，空（2）处

返回 File 类对象的孩子，但 File 类对象没有孩子节点，因此其返回值应该为 NULL。getChildren 方法是继承自抽象父类 AbstractFile，所以其返回类型应该和父类的定义保持一致，空（4）处返回存储孩子节点的集合对象 childList。该程序的运行能够打印出文件目录树，因此空（5）处应该为打印方法的调用。

参考答案

（1）abstract （2）null （3）List

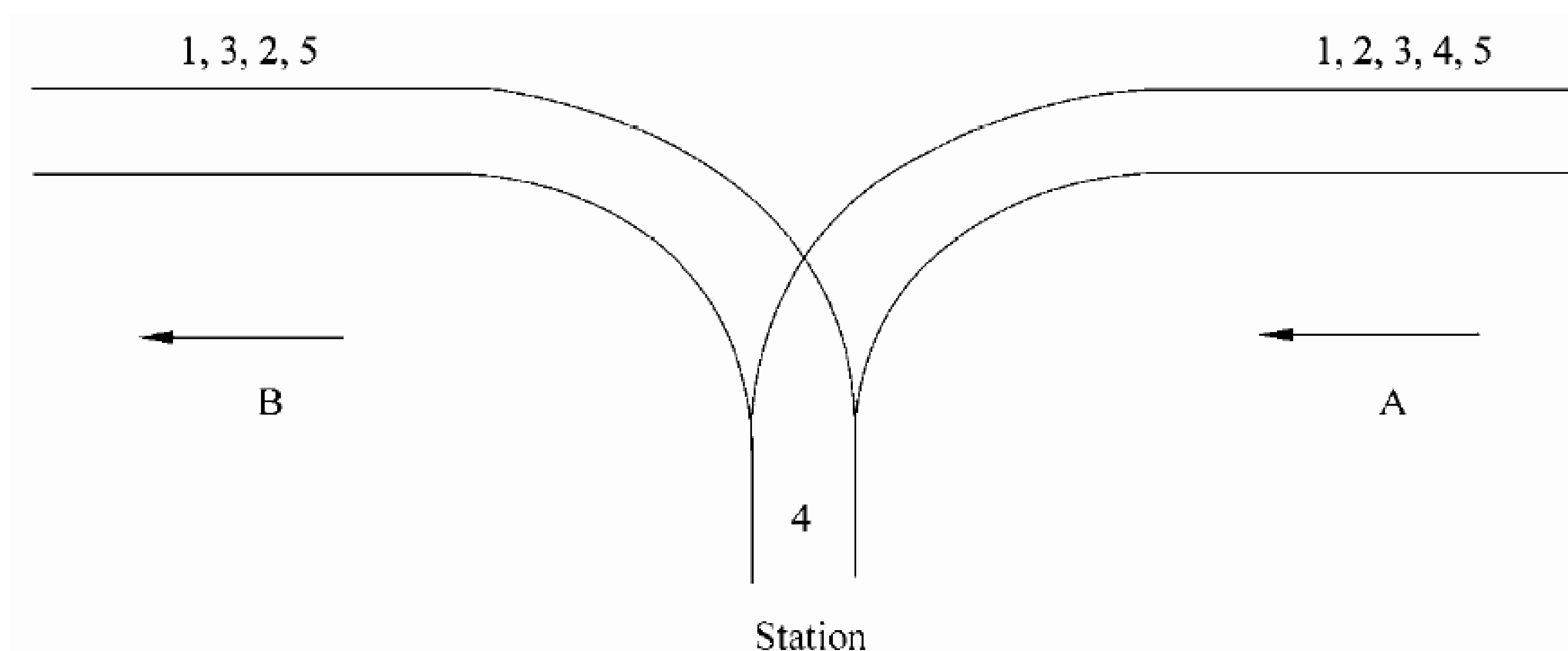
（4）childList （5）printTree(file)

试题七（共 15 分）

阅读以下说明和 C 程序，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

现有 n（n＜1000）节火车车厢，顺序编号为 1，2，3，…，n，按编号连续依次从 A 方向的铁轨驶入，从 B 方向铁轨驶出，一旦车厢进入车站（Station）就不能再回到 A 方向的铁轨上；一旦车厢驶入 B 方向铁轨就不能再回到车站，如下图所示，其中 Station 为栈结构，初始为空且最多能停放 1000 节车厢。



下面的 C 程序判断能否从 B 方向驶出预先指定的车厢序列，程序中使用了栈类型 STACK，关于栈基本操作的函数原型说明如下：

```
void InitStack(STACK *s)：初始化栈
void Push(STACK *s,int e)：将一个整数压栈，栈中元素数目增 1
void Pop(STACK *s)：栈顶元素出栈，栈中元素数目减 1
int Top(STACK s)：返回非空栈的栈顶元素值，栈中元素数目不变
int IsEmpty(STACK s)：若是空栈则返回 1，否则返回 0
```

【C 程序】

```
#include<stdio.h>

/*此处为栈类型及其基本操作的定义，省略*/
```

```
int main( ){
    STACK station;
    int state[1000];
    int n;                              /*车厢数*/
    int begin, i, j, maxNo;             /*maxNo 为 A 端正待入栈的车厢编号*/
    printf("请输入车厢数： ");
    scanf("%d",&n);
    printf("请输入需要判断的车厢编号序列（以空格分隔）： ");
    if (n < 1)  return -1;
    for (i = 0; i<n; i++)   /* 读入需要驶出的车厢编号序列，存入数组 state[] */
      scanf("%d",&state[i]);
    ___(1)___; /*初始化栈*/
    maxNo = 1;
    for(i = 0; i < n;){/*检查输出序列中的每个车厢号 state[i]是否能从栈中获取*/
       if (___(2)___){/*当栈不为空时*/
if (state[i] == Top(station)){ /*栈顶车厢号等于被检查车厢号*/
              printf("%d  ",Top(station));
              Pop(&station);   i++;
          }
          else
             if (___(3)___){
                   printf("error\n");
                   return 1;
             }
             else {
                   begin = ___(4)___;
                   for(j = begin+1; j<=state[i]; j++) {
                         Push(&station, j);
                   }
                 }
       }
       else {  /*当栈为空时*/
             begin = maxNo;
               for(j = begin; j<=state[i]; j++){
                   Push(&station, j);
             }
             maxNo = ___(5)___;
       }
    }
    printf("OK");
```

```
    return 0;
}
```

试题七分析

本题考查栈数据结构的应用和 C 程序设计基本能力。

栈的运算特点是后进先出。在本题中，入栈序列为 1、2、…、n–1、n，出栈序列保存在 state[]数组中，state[0]记录出栈序列的第 1 个元素，state[1]记录出栈序列的第 2 个元素，依此类推。程序采用模拟元素入栈和出栈的操作过程来判断出栈序列是否恰当。需要注意的是，对于栈，应用时不一定是所有元素先入栈后，再逐个进行出栈操作，也不一定是进入一个元素紧接着就出来一个元素，而是栈不满且输入序列还有元素待进入就可以进栈，只要栈不空，栈顶元素就可以出栈，从而使得唯一的一个入栈序列可以得到多个出栈序列。当然，在栈中有多个元素时，只能让栈顶的元素先出栈，栈中其他的元素才能从顶到底逐个出栈。本题中入栈序列和出栈序列的元素为车厢号。

空（1）处对栈进行初始化，根据题干中关于栈基本操作的说明，调用 InitStack 初始化栈，由于形参是指针参数，因此实参应为地址量，即应填入“InitStack(&station)”。

当栈不空时，就可以令栈顶车厢出栈，空（2）处应填入“!IsEmpty(station)”。

栈顶车厢号以 Top(station)表示，若栈顶车厢号等于出栈序列的当前车厢号 state[i]，说明截至目前为止，出栈子序列 state[0]～state[i]可经过栈运算获得。由于进栈时小编号的车厢先于大编号的车厢进入栈中，因此若栈顶车厢号大于出栈序列的当前车厢号 state[i]，则对于 state[i]记录的车厢，若它还在栈中，则此时无法出栈，因为它不在栈顶，所以出错；若它已先于当前的栈顶车厢出栈，则与目前的出栈序列不匹配，仍然出错，因此空（3）处应填入“state[i] < Top(station)”。

若栈顶车厢号小于出栈序列的当前车厢号 state[i]，则说明 state[i]记录的车厢还没有进入栈中，因此从入栈序列（A 端）正待进入的车厢（即比栈顶车厢号正好大 1）开始，直到 state[i]记录的车厢号为止，这些车厢应依次进入栈中。程序中用以下代码实现此操作：

```
for(j = begin+1; j<=state[i]; j++) {
          Push(&station, j);
}
```

显然，begin 应获取栈顶车厢号的值，即空（4）处应填入“Top(station)”。

还有一种情况，就是待考查的出栈序列还没有结束而栈空了，则说明需要处理入栈序列，使其车厢入栈。程序中用 maxNo 表示 A 端正待入栈的车厢编号，相应的处理如下面代码所示：

```
begin = maxNo;
          for(j = begin; j<=state[i]; j++){
```

```
            Push(&station, j);
        }
```

接下来，A 端正待入栈的车厢编号等于 j 或 state[i]+1，即空（5）处应填入 j 或“state[i]+1”。

如果驶出的车厢编号序列是经由栈获得的，则程序运行时输出该序列及字符串“OK”，否则输出“error”而结束。

参考答案

（1）InitStack(&station)　（2）!IsEmpty(station)　（3）state[i] < Top(station)

（4）Top(station)　（5）j，或 state[i]+1

第 5 章　2010 上半年软件设计师上午试题分析与解答

试题（1）

为实现程序指令的顺序执行，CPU__（1）__中的值将自动加 1。

（1）A. 指令寄存器（IR）　　B. 程序计数器（PC）

　　C. 地址寄存器（AR）　　D. 指令译码器（ID）

试题（1）分析

本题考查 CPU 中相关寄存器的基础知识。

指令寄存器（IR）用来保存当前正在执行的指令。当执行一条指令时，先把它从内存取到数据寄存器（DR）中，然后再传送至 IR。为了执行任何给定的指令，必须对操作码进行测试，以便识别所要求的操作。指令译码器（ID）就是做这项工作的。指令寄存器中操作码字段的输出就是指令译码器的输入。操作码一经译码后，即可向操作控制器发出具体操作的特定信号。

地址寄存器（AR）用来保存当前 CPU 所访问的内存单元的地址。由于在内存和 CPU 之间存在着操作速度上的差别，所以必须使用地址寄存器来保持地址信息，直到内存的读/写操作完成为止。

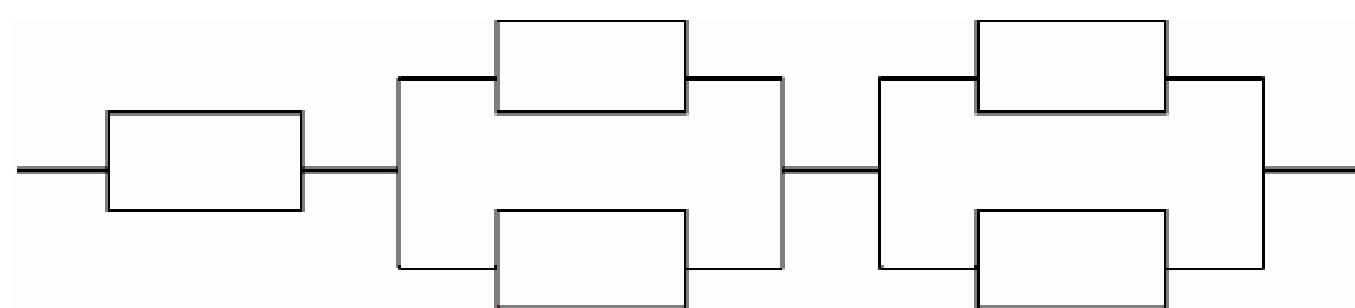
为了保证程序指令能够连续地执行下去，CPU 必须具有某些手段来确定下一条指令的地址。而程序计数器正起到这种作用，所以通常又称为指令计数器。在程序开始执行前，必须将它的起始地址，即程序的一条指令所在的内存单元地址送入 PC，因此程序计数器（PC）的内容即是从内存提取的第一条指令的地址。当执行指令时，CPU 将自动修改 PC 的内容，即每执行一条指令 PC 增加一个量，这个量等于指令所含的字节数，以便使其保持的总是将要执行的下一条指令的地址。由于大多数指令都是按顺序来执行的，所以修改的过程通常只是简单的对 PC 加 1。

参考答案

（1）B

试题（2）

某计算机系统由下图所示的部件构成，假定每个部件的千小时可靠度都为 R，则该系统的千小时可靠度为__（2）__。



（2）A. $R+2R/4$　　B. $R+R^2/4$　　C. $R(1-(1-R)^2)$ D. $R(1-(1-R)^2)^2$

试题（2）分析

本题考查系统可靠性方面的基础知识。

由子系统构成串联系统时，其中任何一个子系统失效就足以使系统失效，其可靠度等于各子系统可靠度的乘积；构成并联系统时，只要有一个子系统正常工作，系统就能正常工作。设每个子系统的可靠性分别以 $R_1, R_2, \cdots, R_N$ 表示，则整个系统的可靠度由下式来求得：

$$R = 1-(1-R_1)(1-R_2)\cdots(1-R_N)$$

因此，本系统的可靠度为 $R(1-(1-R)^2)^2$ 。

参考答案

（2）D

试题（3）

以下关于计算机系统中断概念的叙述中，正确的是＿（3）＿。

（3）A. 由 I/O 设备提出的中断请求和电源掉电都是可屏蔽中断

B. 由 I/O 设备提出的中断请求和电源掉电都是不可屏蔽中断

C. 由 I/O 设备提出的中断请求是可屏蔽中断，电源掉电是不可屏蔽中断

D. 由 I/O 设备提出的中断请求是不可屏蔽中断，电源掉电是可屏蔽中断

试题（3）分析

本题考查中断基础知识。

按照是否可以被屏蔽，可将中断分为两大类：不可屏蔽中断（又叫非屏蔽中断）和可屏蔽中断。不可屏蔽中断源一旦提出请求，CPU 必须无条件响应，而对可屏蔽中断源的请求，CPU 可以响应，也可以不响应。典型的非屏蔽中断源的例子是电源掉电，一旦出现，必须立即无条件地响应，否则进行其他任何工作都是没有意义的。典型的可屏蔽中断源的例子是打印机中断，CPU 对打印机中断请求的响应可以快一些，也可以慢一些，因为让打印机等待是完全可以的。对于软中断，它不受中断允许标志位（IF 位）的影响，所以属于非屏蔽中断范畴。

参考答案

（3）C

试题（4）

与 $\overline{A}\oplus B$ 等价的逻辑表达式是＿（4）＿。（⊕表示逻辑异或，+表示逻辑加）

（4）A. $A+\overline{B}$　　B. $A\oplus\overline{B}$　　C. $A\oplus B$　　D. $AB+\overline{AB}$

试题（4）分析

本题考查逻辑运算基础知识。

用真值表验证如下：

		选项 A	选项 B	选项 C	选项 D	$\overline{A}\oplus B$
A	**B**	$A+\overline{B}$	$A\oplus\overline{B}$	$A\oplus B$	$AB+\overline{AB}$	
0	0	1	1	0	1	1
0	1	0	0	1	0	0
1	0	1	0	1	0	0
1	1	1	1	0	1	1

从上表可知，$\overline{A}\oplus B$ 与 $A\oplus\overline{B}$、$AB+\overline{AB}$ 等价。

参考答案

（4）B 和 D

试题（5）

计算机指令一般包括操作码和地址码两部分，为分析执行一条指令，其__（5）__。

（5）A. 操作码应存入指令寄存器（IR），地址码应存入程序计数器（PC）

B. 操作码应存入程序计数器（PC），地址码应存入指令寄存器（IR）

C. 操作码和地址码都应存入指令寄存器（IR）

D. 操作码和地址码都应存入程序计数器（PC）

试题（5）分析

本题考查指令系统基础知识。

程序被加载到内存后开始运行，当CPU执行一条指令时，先把它从内存储器取到缓冲寄存器DR中，再送入IR暂存，指令译码器根据IR的内容产生各种微操作指令，控制其他的组成部件工作，完成所需的功能。

程序计数器（PC）具有寄存信息和计数两种功能，又称为指令计数器。程序的执行分两种情况，一是顺序执行，二是转移执行。在程序开始执行前，将程序的起始地址送入PC，该地址在程序加载到内存时确定，因此PC的内容即是程序第一条指令的地址。执行指令时，CPU将自动修改PC的内容，以便使其保持的总是将要执行的下一条指令的地址。由于大多数指令都是按顺序来执行的，所以修改的过程通常只是简单地对PC加1。当遇到转移指令时，后继指令的地址根据当前指令的地址加上一个向前或向后转移的位移量得到，或者根据转移指令给出的直接转移地址得到。

参考答案

（5）C

试题（6）

关于 64 位和 32 位微处理器，不能以 2 倍关系描述的是＿（6）＿。

（6）A. 通用寄存器的位数　　B. 数据总线的宽度
　　C. 运算速度　　D. 能同时进行运算的位数

试题（6）分析

本题考查计算机系统基础知识。

计算机系统的运算速度受多种因素的影响，64 位微处理器可同时对 64 位数据进行运算，但不能说其速度是 32 位微处理器的 2 倍。

参考答案

（6）C

试题（7）

Outlook Express 作为邮件代理软件有诸多优点，以下说法中，错误的是＿（7）＿。

（7）A. 可以脱机处理邮件
　　B. 可以管理多个邮件账号
　　C. 可以使用通讯簿存储和检索电子邮件地址
　　D. 不能发送和接收安全邮件

试题（7）分析

Outlook Express 有以下一些优点：

- 可以脱机处理邮件，有效利用联机时间，降低了上网费用。
- 可以管理多个邮件账号，在同一个窗口中使用多个邮件账号。
- 可以使用通讯簿存储和检索电子邮件地址。
- 在邮件中添加个人签名或信纸。
- 发送和接收安全邮件。

参考答案

（7）D

试题（8）、（9）

杀毒软件报告发现病毒 Macro.Melissa，由该病毒名称可以推断病毒类型是＿（8）＿，这类病毒主要感染目标是＿（9）＿。

（8）A. 文件型　　B. 引导型　　C. 目录型　　D. 宏病毒

（9）A. EXE 或 COM 可执行文件　　B. Word 或 Excel 文件
　　C. DLL 系统文件　　D. 磁盘引导区

试题（8）、（9）分析

本题考查计算机病毒方面的基础知识。

计算机病毒的分类方法有许多种，按照最通用的区分方式，即根据其感染的途径以及采用的技术区分，计算机病毒可分为文件型计算机病毒、引导型计算机病毒、宏病毒和目录型计算机病毒。

文件型计算机病毒感染可执行文件（包括 EXE 和 COM 文件）。

引导型计算机病毒影响软盘或硬盘的引导扇区。

目录型计算机病毒能够修改硬盘上存储的所有文件的地址。

宏病毒感染的对象是使用某些程序创建的文本文档、数据库、电子表格等文件，从文件名可以看出 Macro.Melissa 是一种宏病毒，所以题中两空的答案是 D 和 B。

参考答案

（8）D　　（9）B

试题（10）

两个以上的申请人分别就相同内容的计算机程序的发明创造，先后向国务院专利行政部门提出申请，__（10）__可以获得专利申请权。

（10）A. 所有申请人均　　B. 先申请人　　C. 先使用人　　D. 先发明人

试题（10）分析

本题考查知识产权基本知识，即专利管理部门授予专利权的基本原则。我国授予专利权采用先申请原则，即两个以上的申请人分别就同一项发明创造申请专利权的，专利权授予最先申请的人。如果两个以上申请人在同一日分别就同样的发明创造申请专利的，应当在收到专利行政管理部门的通知后自行协商确定申请人。如果协商不成，专利局将驳回所有申请人的申请，即所有申请人均不能取得专利权。所以，先申请人可以获得专利申请权。

参考答案

（10）B

试题（11）

王某是一名程序员，每当软件开发完成后均按公司规定完成软件文档，并上交公司存档，自己没有留存。因撰写论文的需要，王某向公司要求将软件文档原本借出复印，但遭到公司拒绝，理由是该软件文档属于职务作品，著作权归公司。以下叙述中，正确

的是 (11) 。

（11）A. 该软件文档属于职务作品，著作权归公司

B. 该软件文档不属于职务作品，程序员享有著作权

C. 该软件文档属于职务作品，但程序员享有复制权

D. 该软件文档不属于职务作品，著作权由公司和程序员共同享有

试题（11）分析

本题考查知识产权知识，即软件知识产权归属。公民为完成法人或者其他组织工作任务所创作的作品是职务作品。职务作品可以是作品分类中的任何一种形式，如文字作品、电影作品、计算机软件都可能由于为执行工作任务而创作，属于职务作品。其著作权归属分两种情形：

（1）一般职务作品

一般职务作品的著作权由作者享有。单位或其他组织享有在其业务范围内优先使用的权利，期限为 2 年。单位的优先使用权是专有的，未经单位同意，作者不得许可第三人以与单位使用的相同方式使用该作品。在作品完成两年内，如单位在其业务范围内不使用，作者可以要求单位同意由第三人以与单位使用的相同方式使用，所获报酬，由作者与单位按约定的比例分配。

（2）特殊的职务作品

所谓特殊职务作品一是指是利用法人或者其他组织的物质技术条件创作，并由法人或者其他组织承担责任的工程设计、产品设计图纸、地图、计算机软件等职务作品；二是指法律、行政法规规定或者合同约定著作权由法人或者其他组织享有的职务作品。对于特殊职务作品，作者享有署名权，其他权利由法人或非法人单位享有。

本题涉及软件知识产权，王某为完成公司指派的工作任务所开发的软件是职务软件，职务软件包括软件文档和源程序。该软件文档属于特殊职务作品，依据著作权法，对于特殊职务作品，除署名权以外，著作权的其他权利由公司享有。

参考答案

（11）A

试题（12）～（14）

在 ISO 制定并发布的 MPEG 系列标准中， (12) 的音、视频压缩编码技术被应用到 VCD 中， (13) 标准中的音、视频压缩编码技术被应用到 DVD 中， (14) 标准中不包含音、视频压缩编码技术。

（12）A. MPEG-1　　B. MPEG-2　　C. MPEG-7　　D. MPEG-21

（13）A. MPEG-1　　B. MPEG-2　　C. MPEG-4　　D. MPEG-21

（14）A. MPEG-1　　B. MPEG-2　　C. MPEG-4　　D. MPEG-7

试题（12）～（14）分析

MPEG 是 Moving Picture Expert Group 的简称，最初是指由国际标准化组织（ISO）和国际电工委员会（IEC）联合组成的一个研究视频和音频编码标准的专家组。同时 MPEG 也用来命名这个小组所负责开发的一系列音、视频编码标准和多媒体应用标准。这个专家组至今为止已制定和制定中的标准包括 MPEG-1、MPEG-2、MPEG-4、MPEG-7 和 MPEG-21 标准。其中 MPEG-1、MPEG-2 和 MPEG-4 主要针对音、视频编码技术，而 MPEG-7 是多媒体内容描述接口标准，MPEG-21 是多媒体应用框架标准。

VCD 使用了 MPEG-1 标准作为其音、视频信息压缩编码方案，而 MPEG-2 标准中的音、视频压缩编码技术被应用到 DVD 中。

参考答案

（12）A　（13）B　（14）D

试题（15）

基于构件的软件开发，强调使用可复用的软件“构件”来设计和构建软件系统，对所需的构件进行合格性检验、__（15）__，并将它们集成到新系统中。

（15）A. 规模度量　　B. 数据验证　　C. 适应性修改　　D. 正确性测试

试题（15）分析

本题考查基于构件的软件开发基础知识。

基于构件的软件开发，主要强调在构建软件系统时复用已有的软件“构件”，在检索到可以使用的构件后，需要针对新系统的需求对构件进行合格性检验、适应性修改，然后集成到新系统中。

参考答案

（15）C

试题（16）

采用面向对象方法开发软件的过程中，抽取和整理用户需求并建立问题域精确模型的过程叫__（16）__。

（16）A. 面向对象测试　　B. 面向对象实现

C. 面向对象设计　　D. 面向对象分析

试题（16）分析

本题考查面向对象软件开发过程的基础知识。

采用面向对象的软件开发，通常有面向对象分析、面向对象设计、面向对象实现。面向对象分析是为了获得对应用问题的理解，其主要任务是抽取和整理用户需求并建立问题域精确模型。面向对象设计是采用协作的对象、对象的属性和方法说明软件解决方案的一种方式，强调的是定义软件对象和这些软件对象如何协作来满足需求，延续了面向对象分析。面向对象实现主要强调采用面向对象程序设计语言实现系统。面向对象测试是根据规范说明来验证系统设计的正确性。

参考答案

（16）D

试题（17）

使用白盒测试方法时，应根据__（17）__和指定的覆盖标准确定测试数据。

（17）A. 程序的内部逻辑　　B. 程序结构的复杂性
C. 使用说明书　　D. 程序的功能

试题（17）分析

本题考查软件测试方法中白盒测试的基础知识。

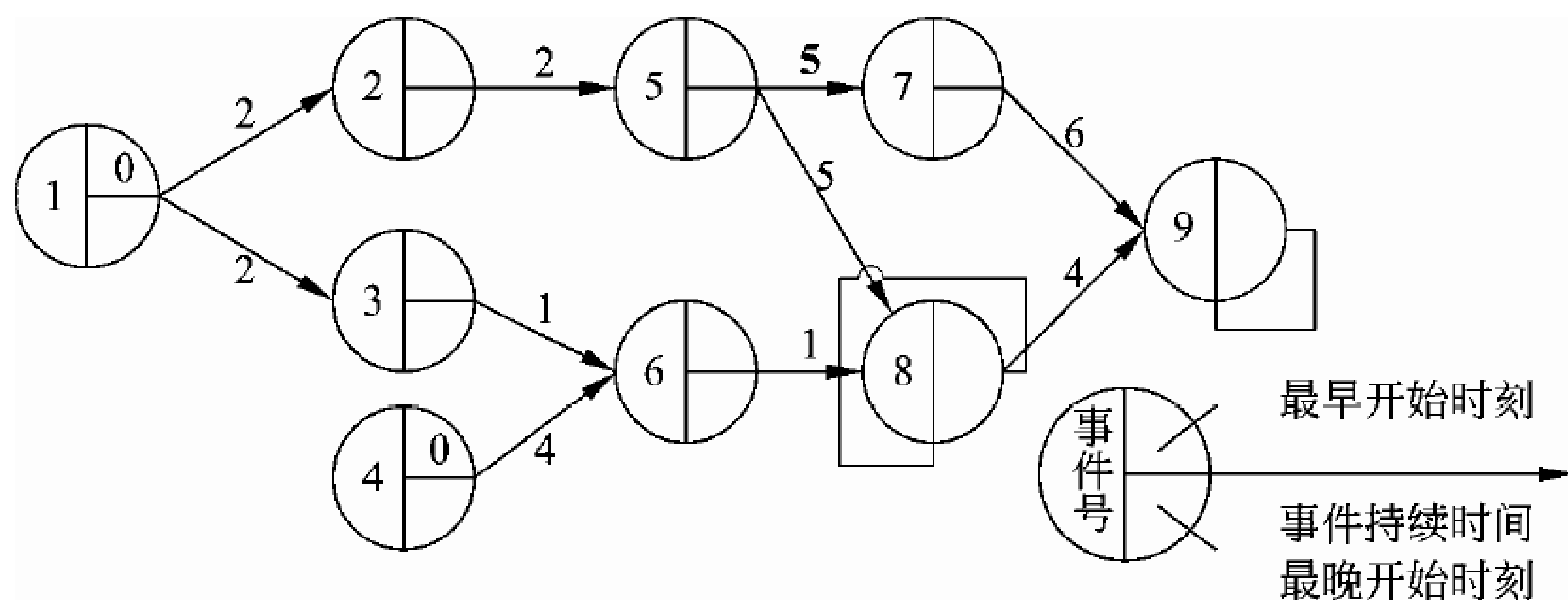
白盒测试也称为结构测试，根据程序的内部结构和逻辑来设计测试用例，对程序的执行路径和过程进行测试，检查是否满足设计的需要。白盒测试常用的技术涉及不同覆盖标准，在测试时需根据指定的覆盖标准确定测试数据。

参考答案

（17）A

试题（18）、（19）

进度安排的常用图形描述方法有 Gantt 图和 PERT 图。Gantt 图不能清晰地描述__（18）__；PERT 图可以给出哪些任务完成后才能开始另一些任务。下图所示的 PERT 图中，事件 6 的最晚开始时刻是__（19）__。



（18）A. 每个任务从何时开始　　B. 每个任务到何时结束
　　C. 每个任务的进展情况　　D. 各任务之间的依赖关系

（19）A. 0　　B. 3　　C. 10　　D. 11

试题（18）、（19）分析

本题考查软件项目计划基础知识。

软件项目计划的一个重要内容是安排进度，常用的方法有 Gantt 图和 PERT 图。Gantt 图用水平条状图描述，它以日历为基准描述项目任务，可以清楚地表示任务的持续时间和任务之间的并行，但是不能清晰地描述各个任务之间的依赖关系。PERT 图是一种网络模型，描述一个项目任务之间的关系。可以明确表达任务之间的依赖关系，即哪些任务完成后才能开始另一些任务，以及如期完成整个工程的关键路径。

图中任务流 1→2→5→7→9 的持续时间是 15，1→2→5→8→9 的持续时间是 13，1→3→6→8→9 的持续时间是 8，4→6→8→9 的持续时间为 9。所以项目关键路径长度为 15。事件 6 在非关键路径上，其后的任务需要时间为 5，所以最晚开始时间=15–5=10。

参考答案

（18）D　（19）C

试题（20）

若某整数的 16 位补码为 $FFFF_H$（H 表示十六进制），则该数的十进制值为＿（20）＿。

（20）A. 0　　B. –1　　C. $2^{16}-1$　　D. $-2^{16}+1$

试题（20）分析

本题考查数据表示基础知识。

根据补码定义，数值 X 的补码记作$[X]_{补}$，如果机器字长为 *n*，则最高位为符号位，0 表示正号，1 表示负号，正数的补码与其原码和反码相同，负数的补码则等于其反码的末尾加 1。

16 位补码能表示的数据范围为$[-2^{15},2^{15}-1]$。对于整数（$2^{16}-1$）和（$-2^{16}+1$），数据表示需要 16 位，再加一个符号位，共 17 位，因此不在 16 位补码能表示的数据范围之内。

在补码表示中，0 有唯一的编码：$[+0]_{补}$0000000000000000，$[-0]_{补}$0000000000000000，即 0000_H。

$[-1]_{原}$=1000000000000001，$[-1]_{反}$=1111111111111110，因此–1 的补码为$[-1]_{补}$=1111111111111111=FFFF。

参考答案

（20）B

试题（21）

逻辑表达式“a∧b∨c∧(b∨x>0)”的后缀式为　（21）　。（其中∧、∨分别表示逻辑与、逻辑或，>表示关系运算大于，对逻辑表达式进行短路求值）

（21）A. abcbx0>∨∧∧∨　　B. ab∧c∨b∧x0>∨

C. ab∧cb∧x>0∨∨　　D. ab∧cbx0>∨∧∨

试题（21）分析

本题考查逻辑表达式的计算及程序语言处理基础知识。

“逻辑与运算”的优先级高于“逻辑或运算”。

“逻辑与运算”表达式“x∧y”的短路求值逻辑是：若 x 为假，则可知“x∧y”的值为假，无须再对 y 求值，因此只有在 x 为真时继续对 y 求值。

“逻辑或运算”表达式“x∨y”的短路求值逻辑是：若 x 为真，则可知“ x ∨y”的值为真，无须再对 y 求值，因此只有在 x 为假时继续对 y 求值。

对于逻辑表达式“a∧b∨c∧(b∨x>0)”，从运算符的优先级方面考虑，需先对“a∧b”求值，然后对“c∧(b∨x>0)”求值，最后进行“∨”运算，因此后缀式为“ab∧cbx0>∨∧∨”。

参考答案

（21）D

试题（22）

编译程序对 C 语言源程序进行语法分析时，可以确定　（22）　。

（22）A. 变量是否定义（或声明）　　B. 变量的值是否正确

C. 循环语句的执行次数　　D. 循环条件是否正确

试题（22）分析

本题考查程序语言基础知识。

对 C 源程序进行编译时，需建立符号表，其作用是记录源程序中各个符号（变量等）的必要信息，以辅助语义的正确性检查和代码生成，在编译过程中需要对符号表进行快速有效地查找、插入、修改和删除等操作。符号表的建立可以始于词法分析阶段，也可以放到语法分析和语义分析阶段，但符号表的使用有时会延续到目标代码的运行阶段。

参考答案

（22）A

试题（23）

如果系统采用信箱通信方式，当进程调用 Send 原语被设置成“等信箱”状态时，其原因是__（23）__。

（23）A. 指定的信箱不存在　　B. 调用时没有设置参数
C. 指定的信箱中无信件　　D. 指定的信箱中存满了信件

试题（23）分析

试题（23）的正确选项为 D。因为 Send 原语是发送原语，如果系统采用信箱通信方式，那么当进程调用 Send 原语被设置成“等信箱”状态时，意味着指定的信箱存满了信件，无可用空间。

参考答案

（23）D

试题（24）

若在系统中有若干个互斥资源 R，6 个并发进程，每个进程都需要 2 个资源 R，那么使系统不发生死锁的资源 R 的最少数目为__（24）__。

（24）A. 6　　B. 7　　C. 9　　D. 12

试题（24）分析

试题（24）的正确选项为 B。对于选项 A，操作系统为每个进程分配 1 个资源 R 后，若这 6 个进程再分别请求 1 个资源 R 时系统已无可供分配的资源 R，则这 6 个进程由于请求的资源 R 得不到满足而死锁。对于选项 B，操作系统为每个进程分配 1 个资源 R 后，系统还有 1 个可供分配的资源 R，能满足其中的 1 个进程的资源 R 要求并运行完毕释放占有的资源 R，从而使其他进程也能得到所需的资源 R 并运行完毕。

参考答案

（24）B

试题（25）、（26）

某进程有 5 个页面，页号为 0～4，页面变换表如下所示。表中状态位等于 0 和 1 分别表示页面不在内存或在内存。若系统给该进程分配了 3 个存储块，当访问的页面 3 不在内存时，应该淘汰表中页号为__（25）__的页面。假定页面大小为 4KB，逻辑地址为十六进制 2C25H，该地址经过变换后，其物理地址应为十六进制__（26）__。

页号	页帧号	状态位	访问位	修改位
0	3	1	1	0
1	—	0	0	0
2	4	1	1	1
3	—	0	0	0
4	1	1	1	1

（25）A. 0　　B. 1　　C. 2　　D. 4

（26）A. 2C25H　　B. 4096H　　C. 4C25H　　D. 8C25H

试题（25）、（26）分析

试题（25）的正确选项为 A。根据题意，页面变换表中状态位等于 0 和 1 分别表示页面不在内存或在内存，所以 0、2 和 4 号页面在内存。当访问的页面 3 不在内存时，系统应该首先淘汰未被访问的页面，因为根据程序的局部性原理，最近未被访问的页面下次被访问的概率更小；如果页面最近都被访问过，应该先淘汰未修改过的页面。因为未修改过的页面内存与辅存一致，故淘汰时无须写回辅存，使系统页面置换代价小。经上述分析，0、2 和 4 号页面都是最近被访问过的，但 2 和 4 号页面都被修改过而 0 号页面未修改过，故应该淘汰 0 号页面。

试题（26）的正确选项为 C。根据题意，页面大小为 4KB，逻辑地址为十六进制 2C25H 其页号为 2，页内地址为 C25H，查页表后可知页帧号（物理块号）为 4，该地址经过变换后，其物理地址应为页帧号 4 拼上页内地址 C25H，即十六进制 4C25H。

参考答案

（25）A　（26）C

试题（27）、（28）

假设某磁盘的每个磁道划分成 9 个物理块，每块存放 1 个逻辑记录。逻辑记录 R0，R1，…，R8 存放在同一个磁道上，记录的安排顺序如下表所示：

物理块	1	2	3	4	5	6	7	8	9
逻辑记录	R0	R1	R2	R3	R4	R5	R6	R7	R8

如果磁盘的旋转速度为 27ms/周，磁头当前处在 R0 的开始处。若系统顺序处理这些记录，使用单缓冲区，每个记录处理时间为 3ms，则处理这 9 个记录的最长时间为<u>（27）</u>；若对信息存储进行优化分布后，处理 9 个记录的最少时间为<u>（28）</u>。

（27）A. 54ms　　B. 108ms　　C. 222ms　　D. 243ms

（28）A. 27ms　　B. 54ms　　C. 108ms　　D. 216ms

试题（27）、（28）分析

试题（27）分析：系统读记录的时间为 27/9＝3ms。对第一种情况：系统读出并处

理记录R1之后，将转到记录R3的开始处，所以为了读出记录R2，磁盘必须再转一圈，需要27ms（转一圈）的时间。这样，处理9个记录的总时间应为222ms。因为处理前8个记录（即R1，R2，…，R8）的总时间再加上读R9时间：8×27ms+ 6ms=222ms。

物理块	1	2	3	4	5	6	7	8	9
逻辑记录	R1	R6	R2	R7	R3	R8	R4	R9	R5

试题（28）分析：对于第二种情况，若对信息进行分布优化的结果如下所示：

从上表可以看出，当读出记录R1并处理结束后，磁头刚好转至R2记录的开始处，立即就可以读出并处理，因此处理9个记录的总时间为：

9×（3ms（读记录）＋3ms（处理记录））＝9×6ms=54ms

参考答案

（27）C （28）B

试题（29）

对于一个大型软件来说，不加控制地变更很快就会引起混乱。为有效地实现变更控制，需借助于配置数据库和基线的概念。__（29）__不属于配置数据库。

（29）A. 开发库 B. 受控库 C. 信息库 D. 产品库

试题（29）分析

本题考查软件变更管理和配置管理的基础知识。

软件变更控制是变更管理的重要内容，要有效进行变更控制，需要借助配置数据库和基线的概念。配置数据库一般包括开发库、受控库和产品库。

参考答案

（29）C

试题（30）

软件设计时需要遵循抽象、模块化、信息隐蔽和模块独立原则。在划分软件系统模块时，应尽量做到__（30）__。

（30）A. 高内聚高耦合 B. 高内聚低耦合

C. 低内聚高耦合 D. 低内聚低耦合

试题（30）分析

本题考查软件设计原则的基础知识。

软件设计时需要遵循抽象、模块化、信息隐蔽和模块独立原则。耦合性和内聚性是模块独立性的两个定性标准，在划分软件系统模块时，尽量做到高内聚、低耦合，提高模块的独立性。

参考答案

（30）B

试题（31）

能力成熟度集成模型 CMMI 是 CMM 模型的最新版本，它有连续式和阶段式两种表示方式。基于连续式表示的 CMMI 共有 6 个（0～5）能力等级，每个能力等级对应到一个一般目标以及一组一般执行方法和特定方法，其中能力等级__（31）__主要关注过程的组织标准化和部署。

（31）A. 1　　B. 2　　C. 3　　D. 4

试题（31）分析

本题考查软件能力成熟度集成模型的基础知识。

能力成熟度集成模型 CMMI 是 CMM 模型的最新版本，基于连续式表述的 CMMI 共有 6 个（0～5）能力等级，对应于未完成级、已执行级、已管理级、已定义级、量化管理级、优化级。每个能力等级对应到一个一般目标，以及一组一般执行方法和特定方法。

能力等级 0 指未执行过程，表明过程域的一个或多个特定目标没有被满足；能力等级 1 指过程通过转化可识别的输入工作产品，产生可识别的输出工作产品，关注于过程域的特定目标的完成；能力等级 2 指过程作为已管理的过程制度化，针对单个过程实例的能力；能力等级 3 指过程作为已定义的过程制度化，关注过程的组织级标准化和部署；能力等级 4 指过程作为定量管理的过程制度化；能力等级 5 指过程作为优化的过程制度化，表明过程得到很好地执行且持续得到改进。

参考答案

（31）C

试题（32）

统一过程（UP）定义了初启阶段、精化阶段、构建阶段、移交阶段和产生阶段，每个阶段以达到某个里程碑时结束，其中__（32）__的里程碑是生命周期架构。

（32）A. 初启阶段　　B. 精化阶段　　C. 构建阶段　　D. 移交阶段

试题（32）分析

本题考查软件开发过程的基础知识。

统一过程（UP）定义了初启阶段、精化阶段、构建阶段、移交阶段和产生阶段，每个阶段达到某个里程碑时结束。其中初启阶段的里程碑是生命周期目标，精化阶段的里程碑是生命周期架构，构建阶段的里程碑是初始运作功能，移交阶段的里程碑是产品发布。

参考答案

（32）B

试题（33）

程序的三种基本控制结构是__（33）__。

（33）A. 过程、子程序和分程序　　B. 顺序、选择和重复

C. 递归、堆栈和队列　　D. 调用、返回和跳转

试题（33）分析

本题考查软件程序设计的基础知识。

程序的三种基本控制结构是顺序结构、选择结构和重复结构。

参考答案

（33）B

试题（34）

__（34）__不属于软件配置管理的活动。

（34）A. 变更标识　　B. 变更控制　　C. 质量控制　　D. 版本控制

试题（34）分析

本题考查软件配置管理的基础知识。

软件配置管理是一组管理整个软件生存期各阶段中变更的活动，主要包括变更标识、变更控制和版本控制。

参考答案

（34）C

试题（35）

一个功能模块M1中的函数F1有一个参数需要接收指向整型的指针，但是在功能模块M2中调用F1时传递了一个整型值，在软件测试中，__（35）__最可能测出这一问题。

（35）A. M1的单元测试　　B. M2的单元测试

C. M1和M2的集成测试　　D. 确认测试

试题（35）分析

本题考查软件测试基础知识。

单元测试侧重于模块中的内部处理逻辑和数据结构，所有模块都通过了测试之后，把模块集成起来仍可能会出现穿越模块的数据丢失、模块之间的相互影响等问题，因此，需要模块按系统设计说明书的要求组合起来进行测试，即集成测试，以发现模块之间协作的问题。

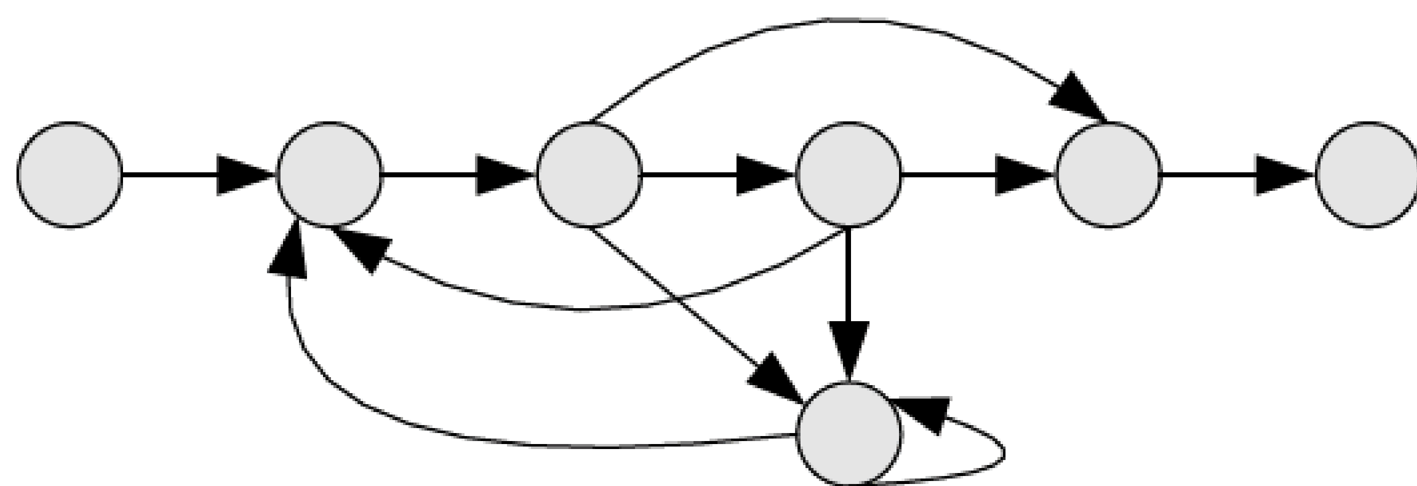
一个功能模块 M1 中的函数 F1 有一个参数需要接收指向整型的指针，但是在功能模块 M2 中调用 F1 时传递了一个整型值，这种模块之间传递参数的错误，在集成测试中最可能测试出来。

参考答案

（35）C

试题（36）

某程序的程序图如下图所示，运用 McCabe 度量法对其进行度量，其环路复杂度是<u>（36）</u>。



（36）A. 4　　B. 5　　C. 6　　D. 8

试题（36）分析

本题考查软件度量的基础知识。

McCabe 度量法是一种基于程序控制流的复杂性度量方法。采用这种方法先画出程序图，然后采用公式 $V(G) = m - n + 2$ 计算环路复杂度。其中，m 是图 G 中弧的个数，n 是图 G 中的结点数。图中结点数为 7，边数为 11，所以环路复杂度为 11–7+2=6。

参考答案

（36）C

试题（37）

以下关于面向对象继承的叙述中，错误的是<u>（37）</u>。

（37）A. 继承是父类和子类之间共享数据和方法的机制

B. 继承定义了一种类与类之间的关系

C. 继承关系中的子类将拥有父类的全部属性和方法

D. 继承仅仅允许单重继承，即不允许一个子类有多个父类

试题（37）分析

本题考查面向对象中继承基本概念。

面向对象技术中，继承是父类和子类之间共享数据和方法的机制。这是类之间的一种关系，在定义和实现一个类的时候，可以在一个已经存在的类的基础上来进行，把这个已经存在的类所定义的内容作为自己的内容，并加入若干新的内容。可以存在多重继承的概念，但不同的程序设计语言可以有自己的规定。

参考答案

（37）D

试题（38）～（40）

不同的对象收到同一消息可以产生完全不同的结果，这一现象叫做（38）。绑定是一个把过程调用和响应调用所需要执行的代码加以结合的过程。在一般的程序设计语言中，绑定是在编译时进行的，叫做（39）；而（40）则是在运行时进行的，即一个给定的过程调用和代码的结合直到调用发生时才进行的。

（38）A. 继承　　B. 多态　　C. 动态绑定　　D. 静态绑定

（39）A. 继承　　B. 多态　　C. 动态绑定　　D. 静态绑定

（40）A. 继承　　B. 多态　　C. 动态绑定　　D. 静态绑定

试题（38）～（40）分析

本题考查面向对象中的基本概念。

在收到消息时，对象要予以响应。不同的对象收到同一消息可以产生完全不同的结果，这一现象叫做多态（polymorphism）。在使用多态的时候，用户可以发送一个通用的消息，而实现的细节则由接收对象自行决定。这样，同一消息就可以调用不同的方法。绑定是一个把过程调用和响应调用所需要执行的代码加以结合的过程。在一般的程序设计语言中，绑定是在编译时进行的，叫做静态绑定。动态绑定则是在运行时进行的，因此，一个给定的过程调用和代码的结合直到调用发生时才进行。

动态绑定是和类的继承以及多态相联系的。在继承关系中，子类是父类的一个特例，所以父类对象可以出现的地方，子类对象也可以出现。因此在运行过程中，当一个对象发送消息请求服务时，要根据接收对象的具体情况将请求的操作与实现的方法进行连接，

即动态绑定。

参考答案

（38）B　（39）D　（40）C

试题（41）

（41）不是面向对象分析阶段需要完成的。

（41）A. 认定对象　　B. 组织对象

C. 实现对象及其相互关系　　D. 描述对象间的相互作用

试题（41）分析

本题考查面向对象分析的基本概念。

面向对象分析包含 5 个活动：认定对象、组织对象、描述对象间的相互作用、定义对象的操作、定义对象的内部信息。

认定对象是指：在应用领域中，按自然存在的实体确立对象。在定义域中，首先将自然存在的“名词”作为一个对象，这通常是研究问题定义域实体的良好开始。通过实体间的关系寻找对象常常没有问题，而困难在于寻找（选择）系统关心的实质性对象。实质性对象是系统稳定性的基础。例如在银行应用系统中，实质性对象应包含客户账务、清算等，而门卫值班表不是实质性对象，甚至可不包含在该系统中。

组织对象含义是：分析对象间的关系，将相关对象抽象成类，其目的是为了简化关联对象，利用类的继承性建立具有继承性层次的类结构。抽象类时可从对象间的操作或一个对象是另一个对象的一部分来考虑；如房子由门和窗构成，门和窗是房子类的子类。由对象抽象类，通过相关类的继承构造类层次，所以说系统的行为和信息间的分析过程是一种迭代表征过程。

描述对象间的相互作用是：描述出各对象在应用系统中的关系。如一个对象是另一个对象的一部分，一个对象与其他对象间的通信关系等。这样可以完整地描述每个对象的环境，由一个对象解释另一个对象，以及一个对象如何生成另一个对象，最后得到对象的界面描述。

实现对象及其相互关系应该归入到系统的实现阶段，不属于分析阶段的任务。

参考答案

（41）C

试题（42）

以下关于面向对象设计的叙述中，错误的是（42）。

（42）A. 面向对象设计应在面向对象分析之前，因为只有产生了设计结果才可对其

进行分析

B. 面向对象设计与面向对象分析是面向对象软件过程中两个重要的阶段

C. 面向对象设计应该依赖于面向对象分析的结果

D. 面向对象设计产生的结果在形式上可以与面向对象分析产生的结果类似，例如都可以使用 UML 表达

试题（42）分析

本题考查面向对象设计与面向对象分析的基本概念。

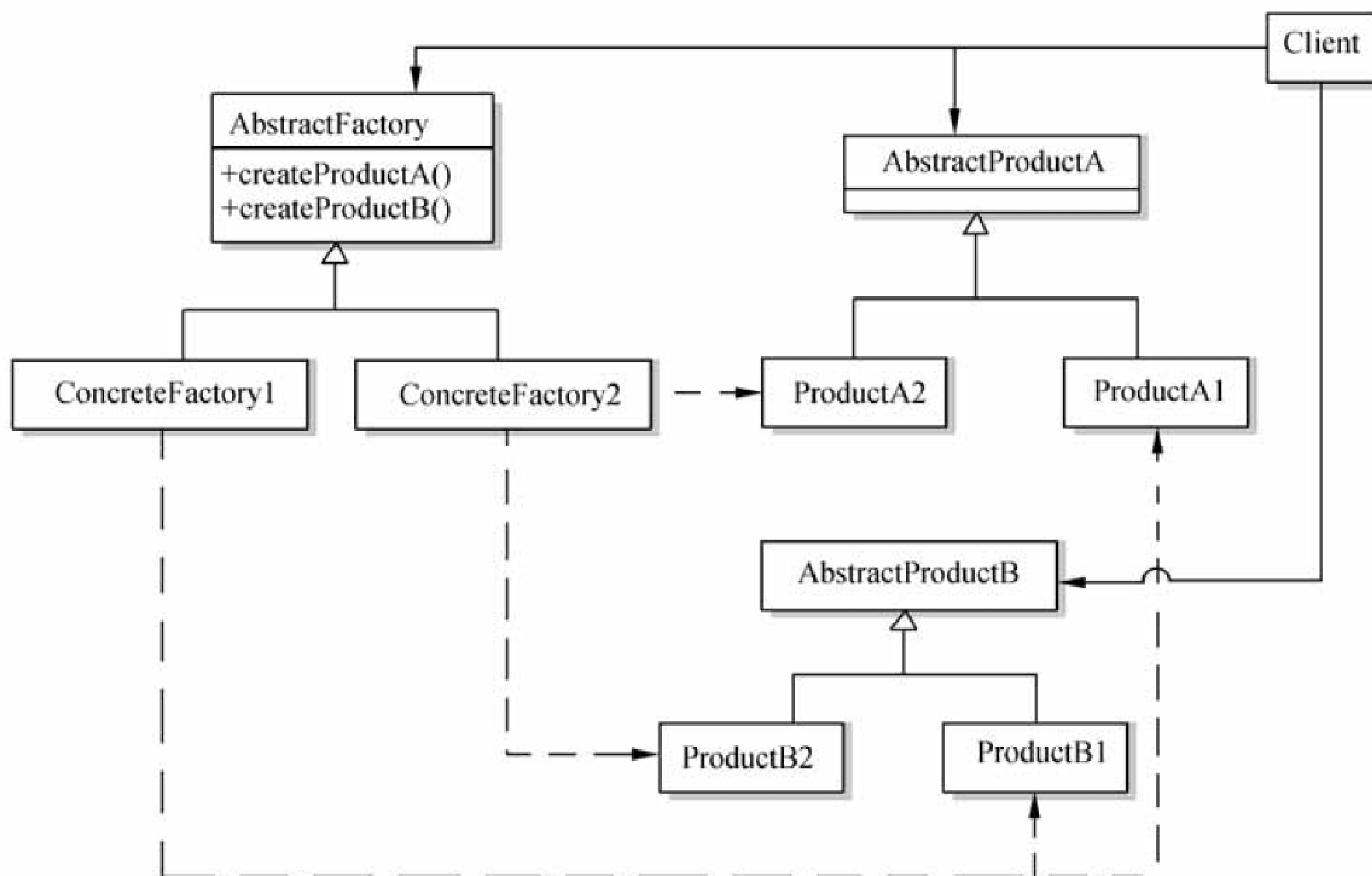
面向对象分析与设计是面向对象软件开发过程中的两个重要阶段，面向对象分析产生分析模型，该分析模型可以使用 UML 表达，面向对象设计以分析模型为基础，继续对分析模型进行精化，得到设计模型，其表达仍然可以采用 UML 建模语言。

参考答案

（42）A

试题（43）、（44）

下列 UML 类图表示的是<u>（43）</u>设计模式。关于该设计模式的叙述中，错误的是<u>（44）</u>。



（43）A. 工厂方法　　B. 策略　　C. 抽象工厂　　D. 观察者

（44）A. 提供创建一系列相关或相互依赖的对象的接口，而无须指定这些对象所属

的具体类

B. 可应用于一个系统要由多个产品系列中的一个来配置的时候

C. 可应用于强调一系列相关产品对象的设计以便进行联合使用的时候

D. 可应用于希望使用已经存在的类，但其接口不符合需求的时候

试题（43）、（44）分析

本题考查面向对象方法中的设计模式知识。

题中的类图是抽象工厂设计模式，该设计模式的意图是提供一个创建一系列相关或相互依赖对象的接口，而无须指定它们具体的类。使用抽象工厂设计模式的常见情形是：一个系统要独立于其产品的创建、组合和表示时；一个系统要由多个产品系列中的一个来配置时；当需要强调一系列相关的产品对象的设计以便进行联合使用时；当提供一个产品类库，而只想显示它们的接口而不是实现时。对于希望使用已经存在的类，但其接口不符合需求的情形，应当考虑桥接设计模式。

参考答案

（43）C　（44）D

试题（45）～（47）

UML 类图中类与类之间的关系有五种：依赖、关联、聚合、组合与继承。若类 A 需要使用标准数学函数类库中提供的功能，那么类 A 与标准类库提供的类之间存在<u>（45）</u>关系；若类 A 中包含了其他类的实例，且当类 A 的实例消失时，其包含的其他类的实例也消失，则类 A 和它所包含的类之间存在<u>（46）</u>关系；若类 A 的实例消失时，其他类的实例仍然存在并继续工作，那么类 A 和它所包含的类之间存在<u>（47）</u>关系。

（45）A. 依赖　B. 关联　C. 聚合　D. 组合

（46）A. 依赖　B. 关联　C. 聚合　D. 组合

（47）A. 依赖　B. 关联　C. 聚合　D. 组合

试题（45）～（47）分析

本题考查 UML 类图之间类与类之间的关系。

UML 类图中类与类之间的关系有五种：依赖、关联、聚合、组合与继承。依赖是几种关系中最弱的一种关系，通常，使用类库就是其中的一种关系。聚合与组合都表示了整体和部分的关系。组合的程度比聚合高，当整体对象消失时，部分对象也随之消失，则属于组合关系，当整体对象消失而部分对象依然可以存在并继续被使用时，则属于聚合关系。

参考答案

（45）A　（46）D　（47）C

试题（48）

以下关于高级语言程序的编译和解释的叙述中，正确的是__（48）__。

（48）A. 编译方式下，可以省略对源程序的词法分析、语法分析
B. 解释方式下，可以省略对源程序的词法分析、语法分析
C. 编译方式下，在机器上运行的目标程序完全独立于源程序
D. 解释方式下，在机器上运行的目标程序完全独立于源程序

试题（48）分析

本题考查语言处理基础知识。

编译和解释是语言处理的两种基本方式。编译过程包括词法分析、语法分析、语义分析、中间代码生成、代码优化和目标代码生成等阶段，以及符号表管理和出错处理模块。

解释过程在词法、语法和语义分析方面与编译程序的工作原理基本相同，但是在运行用户程序时，它直接执行源程序或源程序的内部形式。

这两种语言处理程序的根本区别是：在编译方式下，机器上运行的是与源程序等价的目标程序，源程序和编译程序都不再参与目标程序的执行过程；而在解释方式下，解释程序和源程序（或其某种等价表示）要参与到程序的运行过程中，运行程序的控制权在解释程序。解释器翻译源程序时不产生独立的目标程序，而编译器则需将源程序翻译成独立的目标程序。

参考答案

（48）C

试题（49）

标记语言用一系列约定好的标记来对电子文档进行标记，以实现对电子文档的语义、结构及格式的定义。__（49）__不是标记语言。

（49）A. HTML　　B. XML　　C. WML　　D. PHP

试题（49）分析

本题考查程序语言基础知识。

HTML（Hypertext Marked Language，超文本标记语言），用于互联网的信息表示。用HTML编写的超文本文档称为HTML文档，它能独立于各种操作系统平台（如UNIX，Windows等）。HTML文档是纯文本文档，可以使用记事本、写字板等编辑工具来编写HTML文件，其文件（文档）的扩展名是.html或.htm，它们需要通过WWW浏览器进行解释并显示出效果。

XML（Extensible Markup Language，可扩展的标记语言）1.0标准于1998年2月10日发布，被认为是继HTML和Java编程语言之后的又一个里程碑式的Internet技术。XML

丰富了 HTML 的描述功能，可以描述非常复杂的 Web 页面，如复杂的数字表达式、化学方程式等。XML 的特点是结构化、自描述、可扩展和浏览器自适应等。

用于 WAP 的标记语言就是 WML（Wireless Markup Language），其语法跟 XML 一样，是 XML 的子集。

PHP（Hypertext Preprocessor）是一种在服务器端执行的、嵌入 HTML 文档的脚本语言，其语言风格类似于 C 语言，被网站编程人员广泛运用。

参考答案

（49）D

试题（50）

对于正规式 0*(10*1)*0*，其正规集中字符串的特点是__（50）__。

（50）A. 开头和结尾必须是 0　　B. 1 必须出现偶数次
C. 0 不能连续出现　　D. 1 不能连续出现

试题（50）分析

本题考查程序语言基础知识。

闭包运算符“*”将其运算对象进行若干次连接，因此 0*表示若干个 0 构成的串，而(10*1)*则表示偶数个 1 构成的串。

参考答案

（50）B

试题（51）

确定系统边界和关系规范化分别在数据库设计的__（51）__阶段进行。

（51）A. 需求分析和逻辑设计　　B. 需求分析和概念设计
C. 需求分析和物理设计　　D. 逻辑设计和概念设计

试题（51）分析

试题（51）的正确选项为 A。需求分析阶段的任务是：对现实世界要处理的对象（组织、部门、企业等）进行详细调查，在了解现行系统的概况，确定新系统功能的过程中，确定系统边界、收集支持系统目标的基础数据及其处理方法。

逻辑设计阶段的任务之一是对关系模式进一步的规范化处理。因为生成的初始关系模式并不能完全符合要求，会有数据冗余、更新异常存在，这就需要根据规范化理论对关系模式进行分解，以消除冗余和更新异常。不过有时根据处理要求，可能还需要增加部分冗余以满足处理要求。逻辑设计阶段的任务就需要作部分关系模式的处理，分解、合并或增加冗余属性，提高存储效率和处理效率。

参考答案

（51）A

试题（52）

若关系R、S如下图所示，则关系代数表达式$\pi_{1,3,7}(\sigma_{3<6}(R \times S))$与___（52）___等价。

A	B	C	D
1	2	4	6
2	3	3	1
3	4	1	3

R

C	D	E
3	4	2
8	9	3

S

（52）A. $\pi_{A,C,E}(\sigma_{C<D}(R \times S))$　　B. $\pi_{A,R.C,E}(\sigma_{R.C<S.D}(R \times S))$

C. $\pi_{A,S.C,S.E}(\sigma_{R.C<S.D}(R \times S))$　　D. $\pi_{R.A,R.C,R.E}(\sigma_{R.C<S.D}(R \times S))$

试题（52）分析

试题（52）的正确选项为B。本题考查关系代数运算方面的基础知识。

本题要求关系代数表达式$\pi_{1,3,7}(\sigma_{3<6}(R \times S))$的结果集，其中，R×S的属性列名分别为：R.A，R.B，R.C，R.D，S.C，S.D和S.E，其结果如下表所示：

R.A	R.B	R.C	R.D	S.C	S.D	S.E
1	2	4	6	3	4	2
1	2	4	6	8	9	3
2	3	3	1	3	4	2
2	3	3	1	8	9	3
3	4	1	3	3	4	2
3	4	1	3	8	9	3
	R×S					

$\sigma_{3<6}(R \times S)$的含义是从R×S结果集中选取第三个分量（R.C）小于第六个分量（S.D）的元组，故$\sigma_{3<6}(R \times S)$与$\sigma_{R.C<S.D}(R \times S)$等价。从上表中可以看出，满足条件的结果如下表所示：

R.A	R.B	R.C	R.D	S.C	S.D	S.E
1	2	4	6	8	9	3
2	3	3	1	3	4	2
2	3	3	1	8	9	3
3	4	1	3	3	4	2
3	4	1	3	8	9	3

$\sigma_{3<6}(R \times S)$

$\pi_{1,3,7}(\sigma_{3<6}(R\times S))$ 的含义是从 $\sigma_{3<6}(R\times S)$ 结果集中选取第一列 R.A（或 A）、第三列 R.C 和第七列 S.E（或 E），故 $\pi_{1,3,7}(\sigma_{3<6}(R\times S))$ 与 $\pi_{A,R.C,E}(\sigma_{R.C<S.D}(R\times S))$ 等价。需要说明的是第三列不能简写为 C，因为关系 S 的第一列属性名也为 C，故必须标上关系名加以区别。

参考答案

（52）B

试题（53）～（56）

某销售公司数据库的零件 P（零件号，零件名称，供应商，供应商所在地，库存量）关系如下表所示，其中同一种零件可由不同的供应商供应，一个供应商可以供应多种零件。零件关系的主键为__（53）__。

零件号	零件名称	供应商	供应商所在地	单价（元）	库存量
010023	P2	S1	北京市海淀区 58 号	22.80	380
010024	P3	S1	北京市海淀区 58 号	280.00	1350
010022	P1	S2	陕西省西安市雁塔区 2 号	65.60	160
010023	P2	S2	陕西省西安市雁塔区 2 号	28.00	1280
010024	P3	S2	陕西省西安市雁塔区 2 号	260.00	3900
010022	P1	S3	北京市新城区 65 号	66.80	2860
…	…	…	…	…	…

查询各种零件的平均单价、最高单价与最低单价之间差距的 SQL 语句为：

```
SELECT  零件号，____(54)____
   FROM  P
   ____(55)____;
```

（53）A. 零件号，零件名称　　B. 零件号，供应商
　　C. 零件号，供应商所在地　　D. 供应商，供应商所在地

（54）A. 零件名称，AVG（单价），MAX（单价）−MIN（单价）
　　B. 供应商，AVG（单价），MAX（单价）−MIN（单价）
　　C. 零件名称，AVG 单价，MAX 单价 −MIN 单价
　　D. 供应商，AVG 单价，MAX 单价 −MIN 单价

（55）A. ORDER BY 供应商　　B. ORDER BY 零件号
　　C. GROUP BY 供应商　　D. GROUP BY 零件号

该关系存在冗余以及插入异常和删除异常等问题。为了解决这一问题需要将零件关

系分解为＿（56）＿。

（56）A. P1（零件号，零件名称，单价）、P2（供应商，供应商所在地，库存量）

B. P1（零件号，零件名称）、P2（供应商，供应商所在地，单价，库存量）

C. P1（零件号，零件名称）、P2（零件号，供应商，单价，库存量）、P3（供应商，供应商所在地）

D. P1（零件号，零件名称）、P2（零件号，单价，库存量）、P3（供应商，供应商所在地）、P4（供应商所在地，库存量）

试题（53）～（56）分析

试题（53）的正确选项为 B。根据题意，零件 P 关系的主键为（零件号，供应商）。

试题（54）的正确选项为 A，试题（55）的正确选项为 D。试题要求查询各种零件的平均单价、最高单价与最低单价之间差距，因此，首先需要在结果列中的空（54）填写“零件名称，AVG（单价），MAX（单价）−MIN（单价）”。其次必须用分组语句按零件号分组，故空（55）应填写“GROUP BY 零件号”。完整的 SQL 语句为：

```
SELECT  零件号，零件名称，AVG（单价），MAX（单价）−MIN（单价）
FROM  P
GROUP BY 零件号；
```

试题（56）的正确选项为 C。为了解决关系 P 存在冗余以及插入异常和删除异常等问题，需要将零件关系 P 分解。选项 A、选项 B 和选项 D 是有损连接的，且不保持函数依赖故分解是错误的，例如，分解为选项 A、选项 B 和选项 D 后，用户无法查询某零件由哪些供应商供应，原因是分解是有损连接的，且不保持函数依赖。

参考答案

（53）B　（54）A　（55）D　（56）C

试题（57）

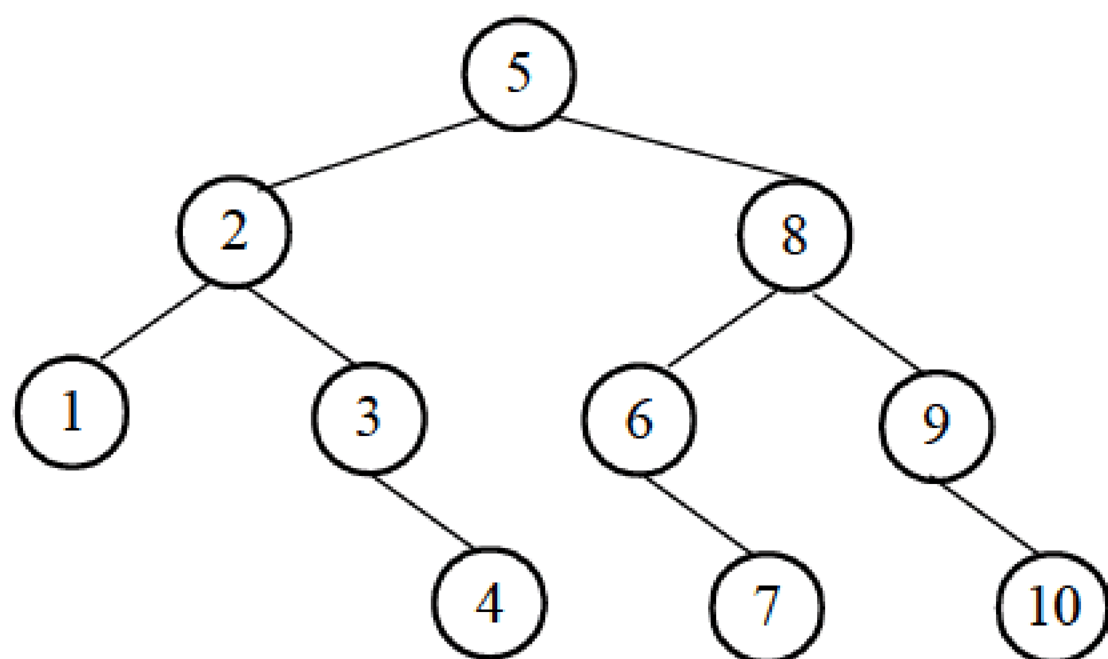
对 n 个元素的有序表 A[1..n]进行二分（折半）查找（除 2 取商时向下取整），查找元素 A[i]（$1 \leqslant i \leqslant n$）时，最多与 A 中的＿（57）＿个元素进行比较。

（57）A. n　　B. $\lfloor \log_2 n \rfloor - 1$　　C. $n/2$　　D. $\lfloor \log_2 n \rfloor + 1$

试题（57）分析

本题考查数据结构基础知识。

二分查找是一种效率较高的查找方法，在 10 个元素构成的有序表中进行二分查找的过程可用二分查找判定树表示，如下图所示：



其中，结点中的数字表示元素在表中的序号。以结点 10 为例，它所在的位置说明若要查找表中的第 10 个元素，则依次与第 5 个、第 8 个、第 9 个和第 10 个元素进行了比较。若有序表中有 n 个元素，则对其进行二分查找的判定树的高度为 $\lfloor \log_2 n \rfloor + 1$（与具有 n 个结点的完全二叉树高度一样），因此，查找过程中最多与 $\lfloor \log_2 n \rfloor + 1$ 个元素进行比较。

参考答案

（57）D

试题（58）

设有如下所示的下三角矩阵 A[0..8,0..8]，将该三角矩阵的非零元素（即行下标不小于列下标的所有元素）按行优先压缩存储在数组 M[1..m]中，则元素 A[i,j]（$0 \leqslant i \leqslant 8, j \leqslant i$）存储在数组 M 的＿（58）＿中。

$$\begin{bmatrix} A_{0,0} & & & & & \\ A_{1,0} & A_{1,1} & & & & \\ . & & . & & 0 & \\ . & & & . & & \\ . & & & & . & \\ A_{7,0} & A_{7,1} & A_{7,2} & \cdots & A_{7,7} & \\ A_{8,0} & A_{8,1} & A_{8,2} & A_{8,3} & \cdots & A_{8,8} \end{bmatrix}$$

（58）A. $M[\frac{i(i+1)}{2}+j+1]$　　B. $M[\frac{i(i+1)}{2}+j]$

C. $M[\frac{i(i-1)}{2}+j]$　　D. $M[\frac{i(i-1)}{2}+j+1]$

试题（58）分析

本题考查数据结构基础知识。

如题图所示，按行方式压缩存储时，A[i,j]之前的元素数目为（1+2+⋯+i+j）个，数

组 M 的下标从 1 开始，因此 A[i,j]的值存储在 $M[\frac{i(i+1)}{2}+j+1]$ 中。

参考答案

（58）A

试题（59）

若用 n 个权值构造一棵最优二叉树（哈夫曼树），则该二叉树的结点总数为__（59）__。

（59）A. $2n$　　B. $2n-1$　　C. $2n+1$　　D. $2n+2$

试题（59）分析

本题考查数据结构基础知识。

二叉树具有以下性质：度为 2 的结点（双分支结点）数比度为 0（叶子结点）数正好少 1。而根据最优二叉树（哈夫曼树）的构造过程可知，最优二叉树中只有度为 2 和 0 的结点，因此，其结点总数为 $2n-1$。

参考答案

（59）B

试题（60）

栈是一种按“后进先出”原则进行插入和删除操作的数据结构，因此，__（60）__必须用栈。

（60）A. 实现函数或过程的递归调用及返回处理时

B. 将一个元素序列进行逆置

C. 链表结点的申请和释放

D. 可执行程序的装入和卸载

试题（60）分析

本题考查数据结构基础知识。

栈是一种后进先出的数据结构。将一个元素序列逆置时，可以使用栈也可以不用。链表结点的申请和释放次序与应用要求相关，不存在“先申请后释放”的操作要求。可执行程序的装入与卸载，也不存在“后进先出”的操作要求。对于函数的递归调用与返回，一定是后被调用执行的先返回。

参考答案

（60）A

试题（61）

对以下四个序列用直接插入排序方法由小到大进行排序时，元素比较次数最少的是

（61）。

（61）A. 89, 27, 35, 78, 41, 15　　B. 27, 35, 41, 16, 89, 70

C. 15, 27, 46, 40, 64, 85　　D. 90, 80, 45, 38, 30, 25

试题（61）分析

本题考查数据结构基础知识。

当序列基本有序时，直接插入排序过程中元素比较的次数较少，当序列为逆序时，元素的比较次数最多。

参考答案

（61）C

试题（62）

对于哈希表，如果将装填因子 α 定义为表中装入的记录数与表的长度之比，那么向表中加入新记录时，（62）。

（62）A. α 的值随冲突次数的增加而递减　　B. α 越大发生冲突的可能性就越大

C. α 等于 1 时不会再发生冲突　　D. α 低于 0.5 时不会发生冲突

试题（62）分析

本题考查数据结构基础知识。

装填因子 α 表示了哈希表的装满程度，显然，α 越大发生冲突的可能性就越大。

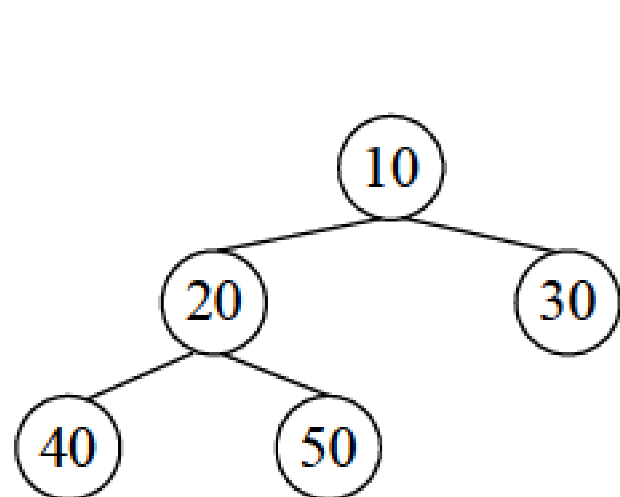
参考答案

（62）B

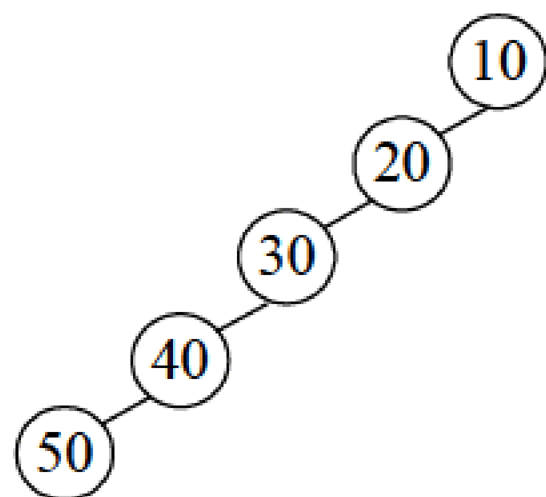
试题（63）

用关键字序列 10、20、30、40、50 构造的二叉排序树（二叉查找树）为（63）。

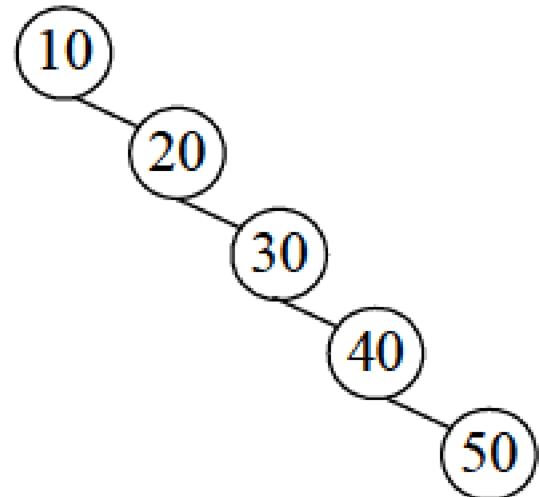
（63）A.



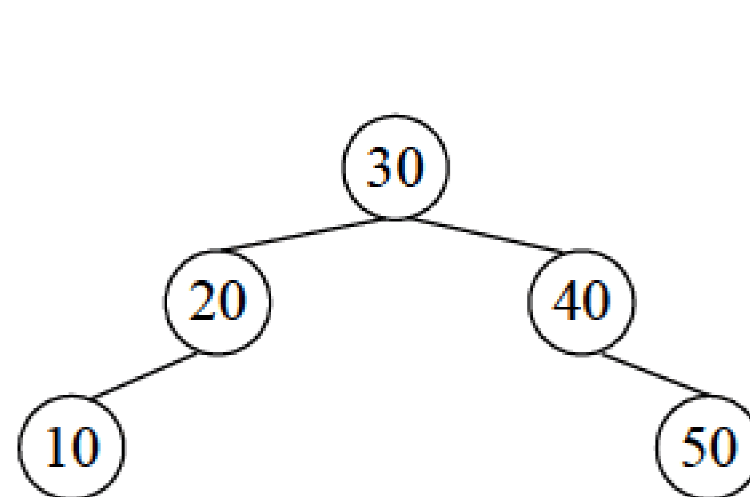
B.



C.



D.



试题（63）分析

本题考查数据结构基础知识。

根据关键字序列构造二叉排序树的基本过程是，若需插入的关键字大于树根，则插入到右子树上，若小于树根，则插入到左子树上，若为空树，则作为树根结点。

参考答案

（63）C

试题（64）

若某算法在问题规模为 n 时，其基本操作的重复次数可由下式表示，则该算法的时间复杂度为 <u>（64）</u> 。

$$T(n)=\begin{cases}1 & n=1\\ T(n-1)+n & n>1\end{cases}$$

（64）A. $O(n)$　　B. $O(n^2)$　　C. $O(\log n)$　　D. $O(n\log n)$

试题（64）分析

本题考查算法分析与设计基础知识。

根据题中给出的递归定义式进行推导，可得 $T(n)=n+n-1+\cdots+2+1$，因此时间复杂度为 $O(n^2)$ 。

参考答案

（64）B

试题（65）

若对一个链表最常用的操作是在末尾插入结点和删除尾结点，则采用仅设尾指针的单向循环链表（不含头结点）时， <u>（65）</u> 。

（65）A. 插入和删除操作的时间复杂度都为 $O(1)$

B. 插入和删除操作的时间复杂度都为 $O(n)$

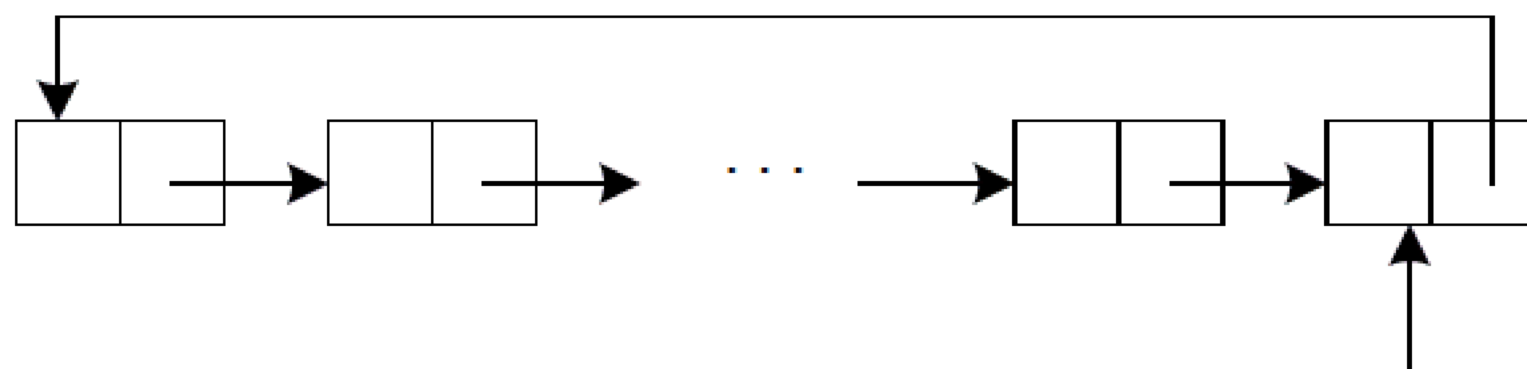
C. 插入操作的时间复杂度为 $O(1)$，删除操作的时间复杂度为 $O(n)$

D. 插入操作的时间复杂度为 $O(n)$，删除操作的时间复杂度为 $O(1)$

试题（65）分析

本题考查数据结构与算法基础知识。

设尾指针的单向循环链表（不含头结点）如下图所示：



设结点的指针域为 next，新结点的指针为 s，则在尾指针所指结点后插入结点的操作为：

```
s->next =t->next; t->next = s; t = s;
```

也就是插入操作的时间复杂度为 $O(1)$。

要删除尾指针所指结点，必须通过遍历操作找到尾结点的前驱结点，其操作序列如下：

```
if (t->next == t) free(t);
else {
    p = t->next;
    while (p->next!=t)
      p = p->next;
    p->next = t->next;
    free(t);
    t = p;
 }
```

也就是说，删除操作的时间复杂度为 $O(n)$。

参考答案

（65）C

试题（66）～（68）

IP 地址块 222.125.80.128/26 包含了 (66) 个可用主机地址，其中最小地址是 (67) ，最大地址是 (68) 。

（66）A. 14　　B. 30　　C. 62　　D. 126

（67）A. 222.125.80.128　　B. 222.125.80.129

　　C. 222.125.80.159　　D. 222.125.80.160

（68）A. 222.125.80.128　　B. 222.125.80.190

　　C. 222.125.80.192　　D. 222.125.80.254

试题（66）～（68）分析

IP 地址块 222.125.80.128/26 留给主机的地址码只有 6 位，$2^6-2=62$。这些地址都采用 222.125.80.10xxxxxx 的形式，其中最小的地址是 222.125.80.10000001，即 222.125.80.129，最大的是 222.125.80.10111110，即 222.125.80.190。

参考答案

（66）C　（67）B　（68）B

试题（69）

以下 HTML 代码中，创建指向邮箱地址的链接正确的是__（69）__。

（69）A. test@test.com

B. test@test.com

C. test@test.com

D. test@test.com

试题（69）分析

本题考查 HTML 语言的基础知识。

在 HTML 语言中，可以通过使用<mailto>标签定义一个指向电子邮件地址的超级链接，通过该链接可以在 Internet 中发送电子邮件。

参考答案

（69）D

试题（70）

POP3 服务默认的 TCP 端口号是__（70）__。

（70）A. 20　B. 25　C. 80　D. 110

试题（70）分析

本试题考查 POP3 服务器的配置。POP3 服务器默认端口为 110，故选 D。

参考答案

（70）D

试题（71）～（75）

Observe that for the programmer, as for the chef, the urgency of the patron（顾客）may govern the scheduled completion of the task, but it cannot govern the actual completion. An omelette（煎鸡蛋）, promised in two minutes, may appear to be progressing nicely. But when it has not set in two minutes, the customer has two choices—waits or eats it raw. Software

customers have had （71） choices.

Now I do not think software （72） have less inherent courage and firmness than chefs, nor than other engineering managers. But false （73） to match the patron's desired date is much more common in our discipline than elsewhere in engineering. It is very （74） to make a vigorous, plausible, and job risking defense of an estimate that is derived by no quantitative method, supported by little data, and certified chiefly by the hunches of the managers.

Clearly two solutions are needed. We need to develop and publicize productivity figures, bug-incidence figures, estimating rules, and so on. The whole profession can only profit from （75） such data. Until estimating is on a sounder basis, individual managers will need to stiffen their backbones and defend their estimates with the assurance that their poor hunches are better than wish derived estimates.

（71）A. no　　B. the same
C. other　　D. lots of

（72）A. testers　　B. constructors
C. managers　　D. architects

（73）A. tasks　　B. jobs
C. works　　D. scheduling

（74）A. easy　　B. difficult
C. simple　　D. painless

（75）A. sharing　　B. excluding
C. omitting　　D. ignoring

参考译文

观察一下编程人员，你可能会发现，同厨师一样，某项任务的计划进度，可能受限于顾客要求的紧迫程度，但紧迫程度无法控制实际的完成情况。就像约好在两分钟内完成一个煎蛋，看上去可能进行得非常好。但当它无法在两分钟内完成时，顾客只能选择等待或者生吃煎蛋。软件顾客的情况类似。

我现在并不认为软件经理内在的勇气和坚持不如厨师，或者不如其他工程经理。但为了满足顾客期望的日期而造成的不合理进度安排，在软件领域中却比其他的任何工程领域要普遍得多。而且，非量化方法的采用，少得可怜的数据支持，加上完全借助软件经理的直觉，这样的方式很难生产出健壮可靠和规避风险的估计。

显然我们需要两种解决方案。开发并推行生产率图表、缺陷率、估算规则等等，整个组织最终会从这些数据的共享上获益。或者在基于可靠基础的估算出现之前，项目经理需要挺直腰杆并坚持他们的估计，确信自己的经验和直觉总比从期望得出的估计要强得多。

参考答案

（71）B　　（72）C　　（73）D　　（74）B　　（75）A

第 6 章　2010 上半年软件设计师下午试题分析与解答

试题一（共 15 分）

阅读下列说明和图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

【说明】

某大型企业的数据中心为了集中管理、控制用户对数据的访问并支持大量的连接需求，欲构建数据管理中间件，其主要功能如下：

（1）数据管理员可通过中间件进行用户管理、操作管理和权限管理。用户管理维护用户信息，用户信息（用户名、密码）存储在用户表中；操作管理维护数据实体的标准操作及其所属的后端数据库信息，标准操作和后端数据库信息存放在操作表中；权限管理维护权限表，该表存储用户可执行的操作信息。

（2）中间件验证前端应用提供的用户信息。若验证不通过，返回非法用户信息；若验证通过，中间件将等待前端应用提交操作请求。

（3）前端应用提交操作请求后，中间件先对请求进行格式检查。如果格式不正确，返回格式错误信息；如果格式正确，则进行权限验证（验证用户是否有权执行请求的操作），若用户无权执行该操作，则返回权限不足信息，否则进行连接管理。

（4）连接管理连接相应的后台数据库并提交操作。连接管理先检查是否存在空闲的数据库连接，如果不存在，新建连接；如果存在，则重用连接。

（5）后端数据库执行操作并将结果传给中间件，中间件对收到的操作结果进行处理后，将其返回给前端应用。

现采用结构化方法对系统进行分析与设计，获得如图 1-1 所示的顶层数据流图和图 1-2 所示的 0 层数据流图。

【问题 1】（3 分）

使用说明中的词语，给出图 1-1 中的实体 E1～E3 的名称。

【问题 2】（3 分）

使用说明中的词语，给出图 1-2 中的数据存储 D1～D3 的名称。

【问题 3】（6 分）

给出图 1-2 中加工 P 的名称及其输入、输出流。

	名　称	起　点	终　点
输入流			P
输出流		P	

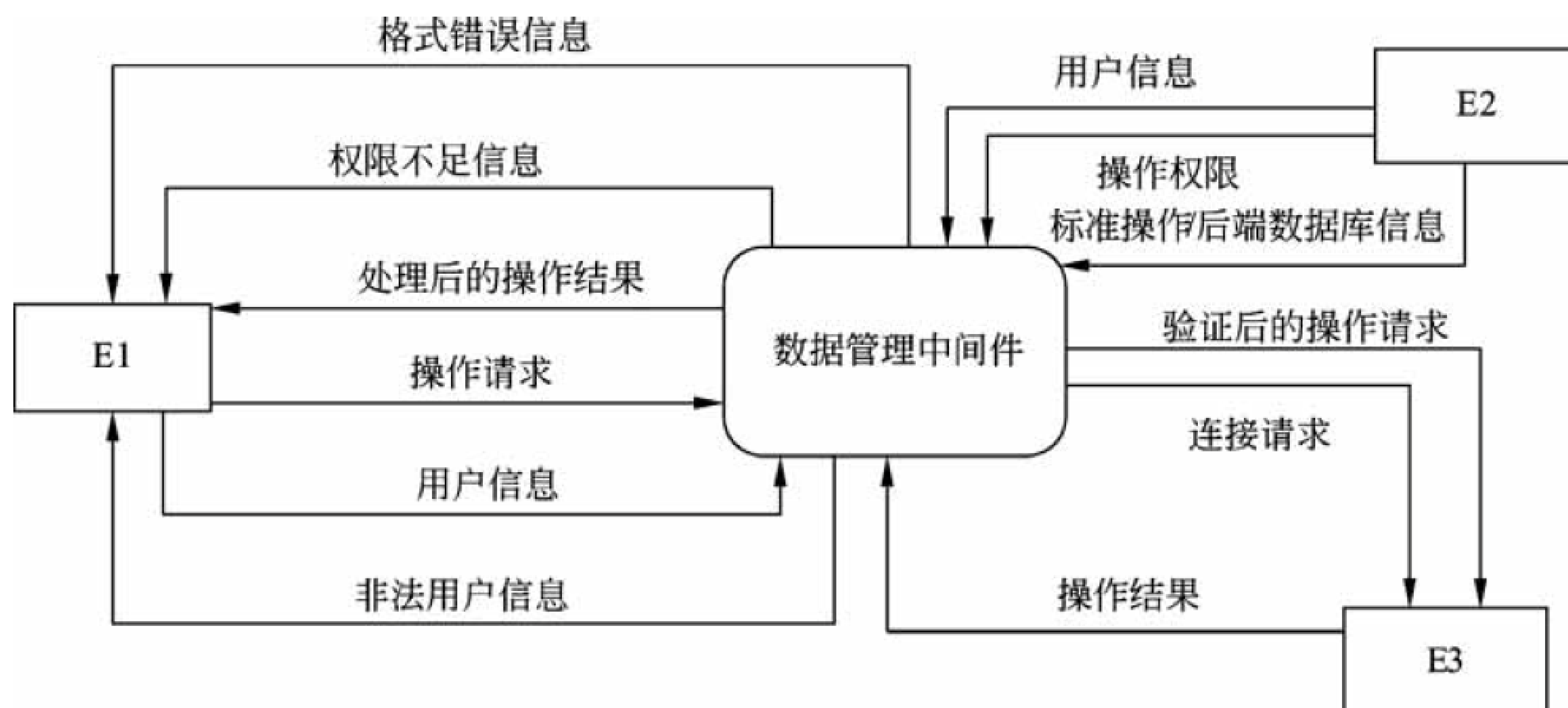


图 1-1　顶层数据流图

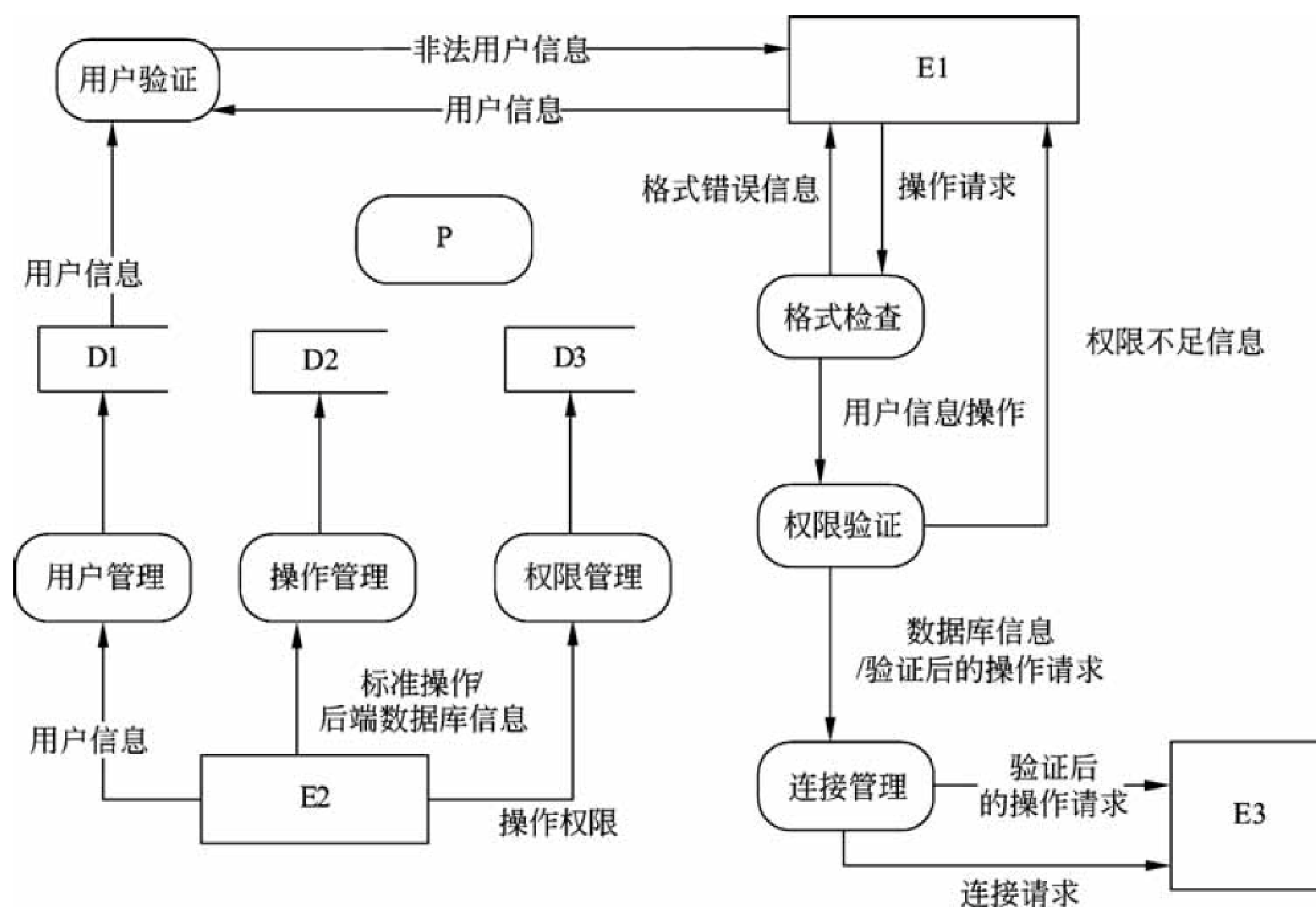


图 1-2　0 层数据流图

除加工 P 的输入与输出流外，图 1-2 还缺失了两条数据流，请给出这两条数据流的起点和终点。

起　点	终　点

注：名称使用说明中的词汇，起点和终点均使用图 1-2 中的符号或词汇。

【问题 4】（3 分）

在绘制数据流图时，需要注意加工的绘制。请给出三种在绘制加工的输入、输出时可能出现的错误。

试题一分析

本题考查数据流图（DFD）的应用，是比较传统的题目，要求考生细心分析题目中所描述的内容。

DFD 是一种便于用户理解、分析系统数据流程的图形工具。是系统逻辑模型的重要组成部分。

【问题 1】

本问题考查顶层 DFD。顶层 DFD 一般用来确定系统边界，将待开发系统看作一个加工，因此图中只有唯一的一个加工和一些外部实体，以及这两者之间的输入输出数据流。题目要求根据描述确定图中的外部实体。分析题目中的描述，并结合已经在顶层数据流图中给出的数据流进行分析。题目中有信息描述：数据管理员可通过中间件进行用户管理、操作管理和权限管理；前端应用提交操作请求；连接管理连接相应的后台数据库并提交操作。由此可知该中间件系统有数据管理员、前端应用和后端数据库三个外部实体。从图 1-1 中数据流和实体的对应关系可知，E1 为前端应用，E2 为数据管理员，E3 为后端数据库。

【问题 2】

本问题考查 0 层 DFD 中数据存储的确定。说明中描述：用户信息（用户名、密码）存储在用户表中；标准操作和后端数据库信息存放在操作表中；权限管理维护信息存放在权限表中。因此数据存储为用户表、操作表以及权限表。再根据图 1-2 可知 D1 的输入数据流从用户管理来，D2 的输入数据流从操作管理来，D3 的输入数据流从权限管理来，所以 D1 为用户表，D2 为操作表，D3 为权限表。

【问题 3】

本问题考查 0 层 DFD 中缺失的加工和数据流。比较图 1-1 和图 1-2，可知顶层 DFD 中的操作结果和处理后的操作结果没有在 0 层 DFD 中体现。再根据描述“后端数据库执行操作并将结果传给中间件，中间件对收到的操作结果进行处理后，将其返回给前端

应用”可知，需要有操作结果处理，因此 P 为操作结果处理，其输入流为从后端数据库 E3 来的操作结果，输出结果为处理后的操作结果，并返回给前端应用 E1。

考查完 P 及其输入输出流之后，对图 1-2 的内部数据流进行考查，以找出缺失的另外 2 条数据流。从图中可以看出 D2 和 D3 只有输入流没有输出流，这是常见 DFD 设计时的错误，所以首先考查 D2 和 D3 的输出流。描述中有“权限验证是验证用户是否有权执行请求的操作，若用户有权执行该操作，进行连接管理；连接管理连接相应的后台数据库并提交操作；权限表存储用户可执行的操作信息”。因此，权限验证有从权限表 D3 来的输入数据流。而要连接后端数据库，需要数据库信息，从权限验证的输出流中包含有数据库信息可知，权限验证需要获取到数据库信息，所以还需从操作表 D2 来的数据流。

【问题 4】

本问题考查在绘制数据流图中加工绘制时的注意事项。绘制加工时可能出现的错误有：加工的输入、输出时可能出现只有输入而无输出、只有输出而无输入、输入的数据流无法通过加工产生输出流以及输入的数据流与输出的数据流名称相同等错误。

参考答案

【问题 1】

E1：前端应用　　　E2：数据管理员　　　E3：后端数据库

【问题 2】

D1：用户表　　　D2：操作表　　　D3：权限表

【问题 3】

P 的名称：操作结果处理

	名　称	起　点	终　点
输入流	操作结果	E3	P
输出流	处理后的操作结果	P	E1

缺少的数据流：

起　点	终　点
D2	权限验证
D3	权限验证

【问题 4】

在绘制数据流图的加工时，可能出现的输入、输出错误：

只有输入而无输出　或者　黑洞

只有输出而无输入　或者　奇迹

输入的数据流无法通过加工产生输出流 或者 灰洞

输入的数据流与输出的数据流名称相同

试题二（共 15 分）

阅读下列说明，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某学校拟开发一套实验管理系统，对各课程的实验安排情况进行管理。

【需求分析】

一个实验室可进行多种类型不同的实验。由于实验室和实验员资源有限，需根据学生人数分批次安排实验室和实验员。一门课程可以为多个班级开设，每个班级每学期可以开设多门课程。一门课程的一种实验可以根据人数、实验室的可容纳人数和实验类型，分批次开设在多个实验室的不同时间段。一个实验室的一次实验可以分配多个实验员负责辅导实验，实验员给出学生的每次实验成绩。

（1）课程信息包括：课程编号、课程名称、实验学时、授课学期和开课的班级等信息；实验信息记录该课程的实验进度信息，包括：实验名、实验类型、学时、安排周次等信息，如表 2-1 所示。

表 2-1　课程及实验信息

课程编号	15054037	课程名称	数字电视原理		实验学时	12
班级	电 0501,信 0501,计 0501	授课院系	机械与电气工程		授课学期	第三学期
序号	实验名		实 验 类	难度	学时	安排周次
1505403701	音视频 AD-DA 实验		验证性	1	2	3
1505403702	音频编码实验		验证性	2	2	5
1505403703	视频编码实验		演示性	0.5	1	9

（2）以课程为单位制定实验安排计划信息，包括：实验地点，实验时间、实验员等信息，实验计划如表 2-2 所示。

表 2-2　实验安排计划

课程编号	15054037	课程名称	数字电视原理	安排学期	2009 年秋	总人数	220
实验编号	实验名		实验员	实验时间	地点	批次号	人数
1505403701	音视频 AD-DA 实验		盛×，陈×	第 3 周周四晚上	实验三楼 310	1	60
1505403701	音视频 AD-DA 实验		盛×，陈×	第 3 周周四晚上	实验三楼 310	2	60
1505403701	音视频 AD-DA 实验		吴×，刘×	第 3 周周五晚上	实验三楼 311	3	60
1505403701	音视频 AD-DA 实验		吴×	第 3 周周五晚上	实验三楼 311	4	40
1505403702	音频编码实验		盛×，刘×	第 5 周周一下午	实验四楼 410	1	70

（3）由实验员给出每个学生每次实验的成绩，包括：实验名、学号、姓名、班级、

实验成绩等信息，实验成绩如表2-3所示。

表2-3 实验成绩

实验员：盛×

实验名	音视频AD-DA实验	课程名	数字电视原理
学号	姓名	班级	实验成绩
030501001	陈民	信0501	87
030501002	刘志	信0501	78
040501001	张勤	计0501	86

（4）学生的实验课程总成绩根据每次实验的成绩以及每次实验的难度来计算。

【概念模型设计】

根据需求阶段收集的信息，设计的实体联系图（不完整）如图2-1所示。

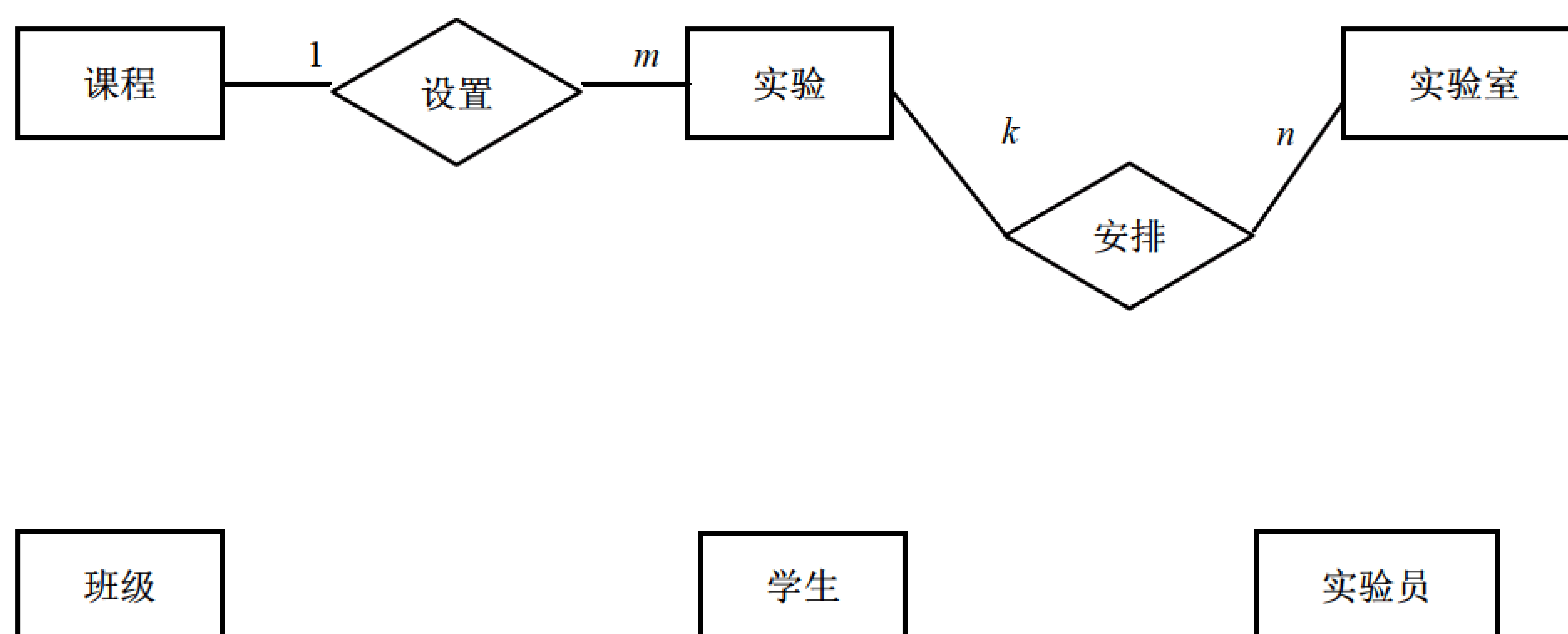


图2-1 实体联系图

【逻辑结构设计】

根据概念模型设计阶段完成的实体联系图，得出如下关系模式（不完整）：

课程（课程编号，课程名称，授课院系，实验学时）
班级（班级号，专业，所属系）
开课情况（____（1）____，授课学期）
实验（____（2）____，实验类型，难度，学时，安排周次）
实验计划（____（3）____，实验时间，人数）
实验员（____（4）____，级别）
实验室（实验室编号，地点，开放时间，可容纳人数，实验类型）
学生（____（5）____，姓名，年龄，性别）
实验成绩（____（6）____，实验成绩，评分实验员）

【问题 1】（6 分）

补充图 2-1 中的联系和联系的类型。

【问题 2】（6 分）

根据图 2-1，将逻辑结构设计阶段生成的关系模式中的空（1）～（6）补充完整并用下划线指出这六个关系模式的主键。

【问题 3】（3 分）

如果需要记录课程的授课教师，新增加“授课教师”实体。请对图 2-1 进行修改，画出修改后的实体间联系和联系的类型。

试题二分析

本题考查数据库概念结构设计及向逻辑结构转换的掌握。

此类题目要求考生认真阅读题目，根据题目的需求描述，给出实体间的联系。

【问题 1】

根据题意，由“一门含实验的课程可以开设给多个班级，每个班级每学期可以开设多门含实验的课程”可知课程和班级之间的开设关系为 $m:n$ 联系。由“一个实验室的一次实验可以分配多个实验员负责辅导实验”可知实验、实验室与实验员之间的安排关系为 $k:n:m$ 联系。由“实验员给出学生的每次实验成绩”可知实验、学生与实验员之间的成绩关系为 $k:n:m$ 联系。班级和学生之间的包含关系为 $1:n$ 联系。

【问题 2】

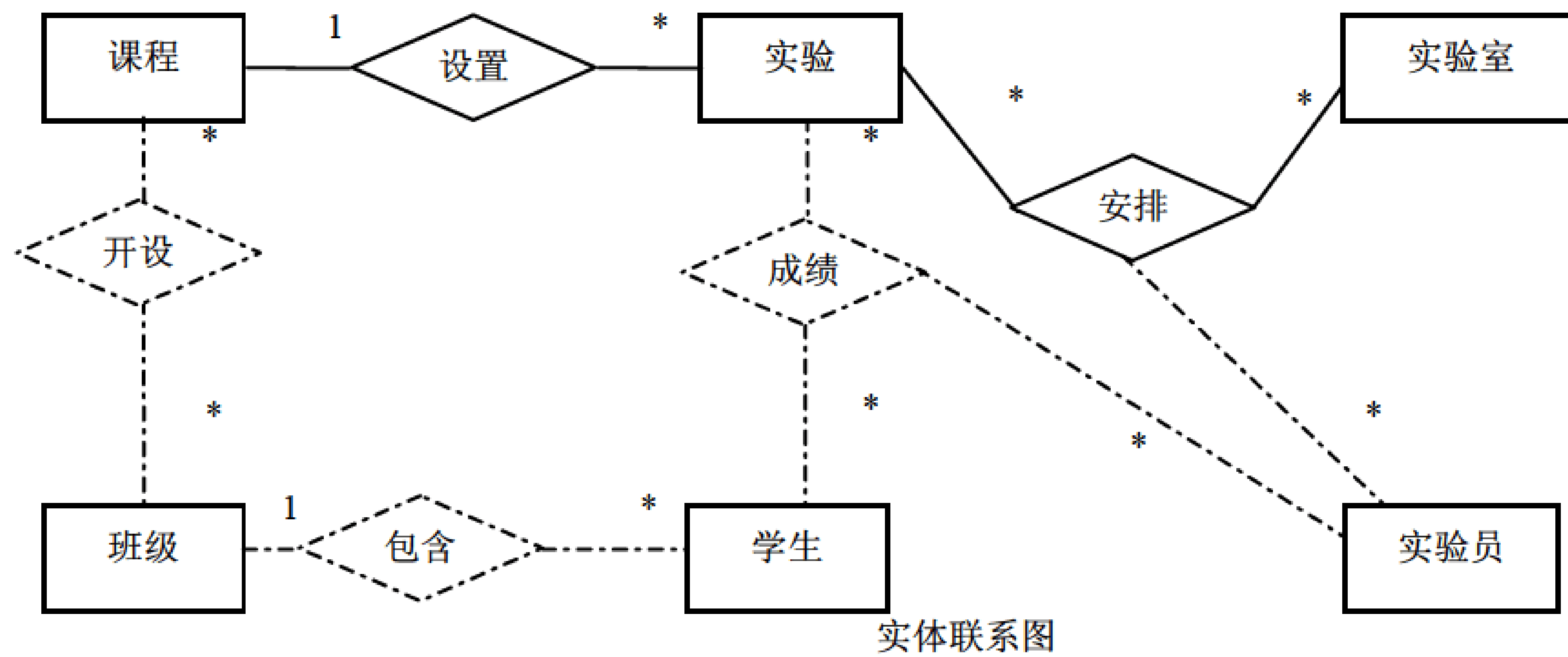
根据题意可知课程编号是课程的主键，班级号是班级的主键。从表 2-1 可知，开课情况是体现课程与班级间的 $m:n$ 联系，因此开课情况关系模式应该包含课程编号和班级号，并共同作为主键。一门课程包含多次实验，实验与课程之间是 $m:1$ 关系，因此，根据表 2-1，实验关系模式应包含实验编号和课程编号，并且以实验编号为主键，以课程编号为外键。在制定试验计划时，每个班的每次实验可能按实验室被分成多个批次，每个批次的实验会有若干名实验员来辅导学生实验并打分。实验员关系模式应该记录实验员编号和实验员姓名，并以实验员编号为主键。实验室编号是实验室的主键。从表 2-2 可见，实验计划关系模式应记录实验编号、批次号和授课学期，并且共同作为主键。从表 2-3 可见，实验成绩关系模式记录每个学生的每次实验成绩，应包含学号和实验编号，并共同作为主键。

【问题 3】

由于授课教师负责给若干个班级开设若干门课程，因此，课程、班级和授课教师之间的开设关系是 $k:n:m$ 联系。

参考答案

【问题 1】



【问题 2】

（1）课程编号，班级号

（2）实验编号，课程编号

（3）实验编号，批次号，安排学期，实验室编号，实验员编号

（4）实验员编号，实验员姓名

（5）学号，班级号

（6）实验编号，学号

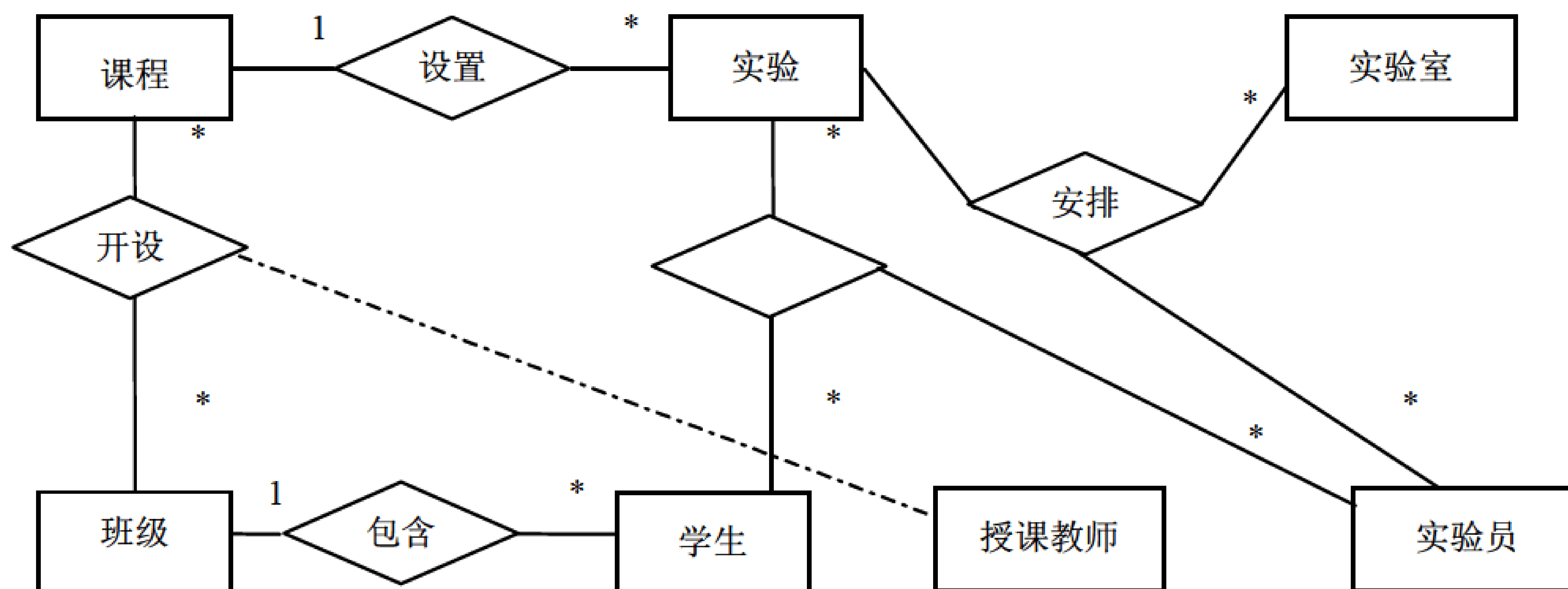
其他关系模式主键：

课程（课程编号，课程名称，授课院系，实验学时）

班级（班级号，专业，所属系）

实验室（实验室编号，地点，开放时间，可容纳人数，实验课类型）

【问题 3】



试题三（共 15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某运输公司决定为新的售票机开发车票销售的控制软件。图 3-1 给出了售票机的面板示意图以及相关的控制部件。

售票机相关部件的作用如下所述：

（1）目的地键盘用来输入行程目的地的代码（例如，200 表示总站）。

（2）乘客可以通过车票键盘选择车票种类（单程票、多次往返票和座席种类）。

（3）继续/取消键盘上的取消按钮用于取消购票过程，继续按钮允许乘客连续购买多张票。

（4）显示屏显示所有的系统输出和用户提示信息。

（5）插卡口接受 MCard（现金卡），硬币口和纸币槽接受现金。

（6）打印机用于输出车票。

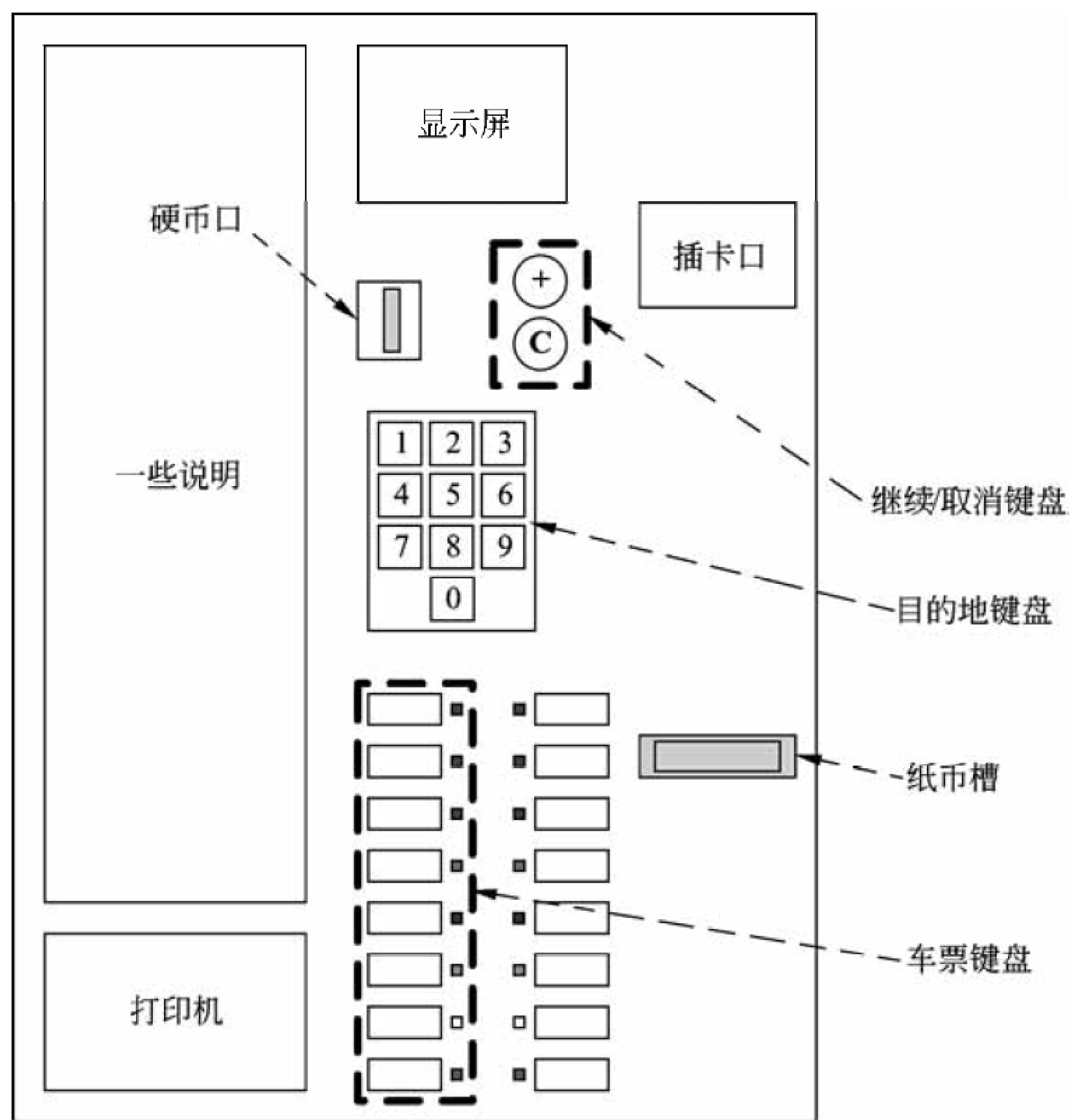


图 3-1　售票机面板示意图

假设乘客总是支付恰好需要的金额而无须找零，售票机的维护工作（取回现金、放入空白车票等）由服务技术人员完成。

系统采用面向对象方法开发，使用 UML 进行建模。系统的顶层用例图和类图分别如图 3-2 和图 3-3 所示。

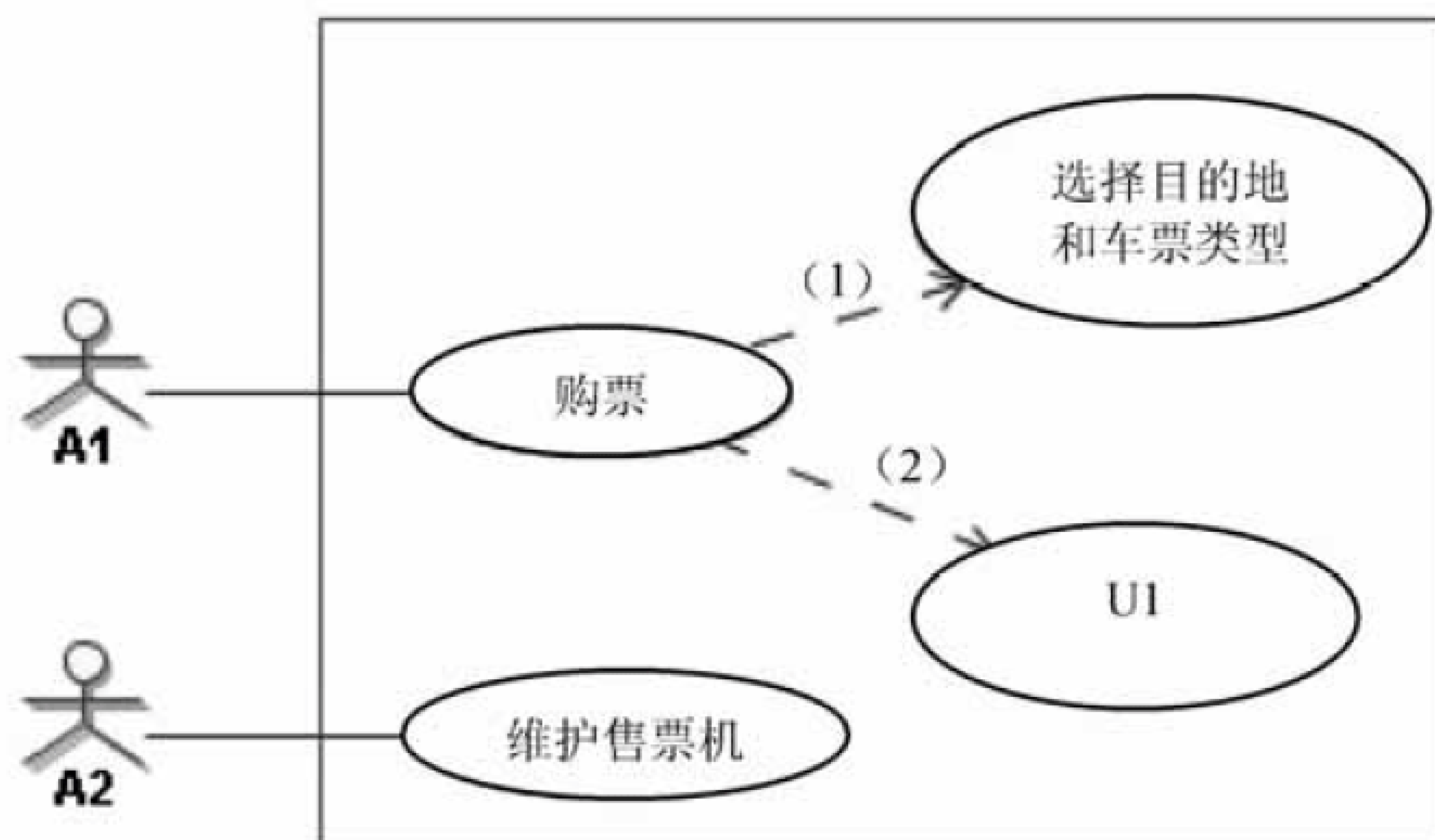


图 3-2 顶层用例图

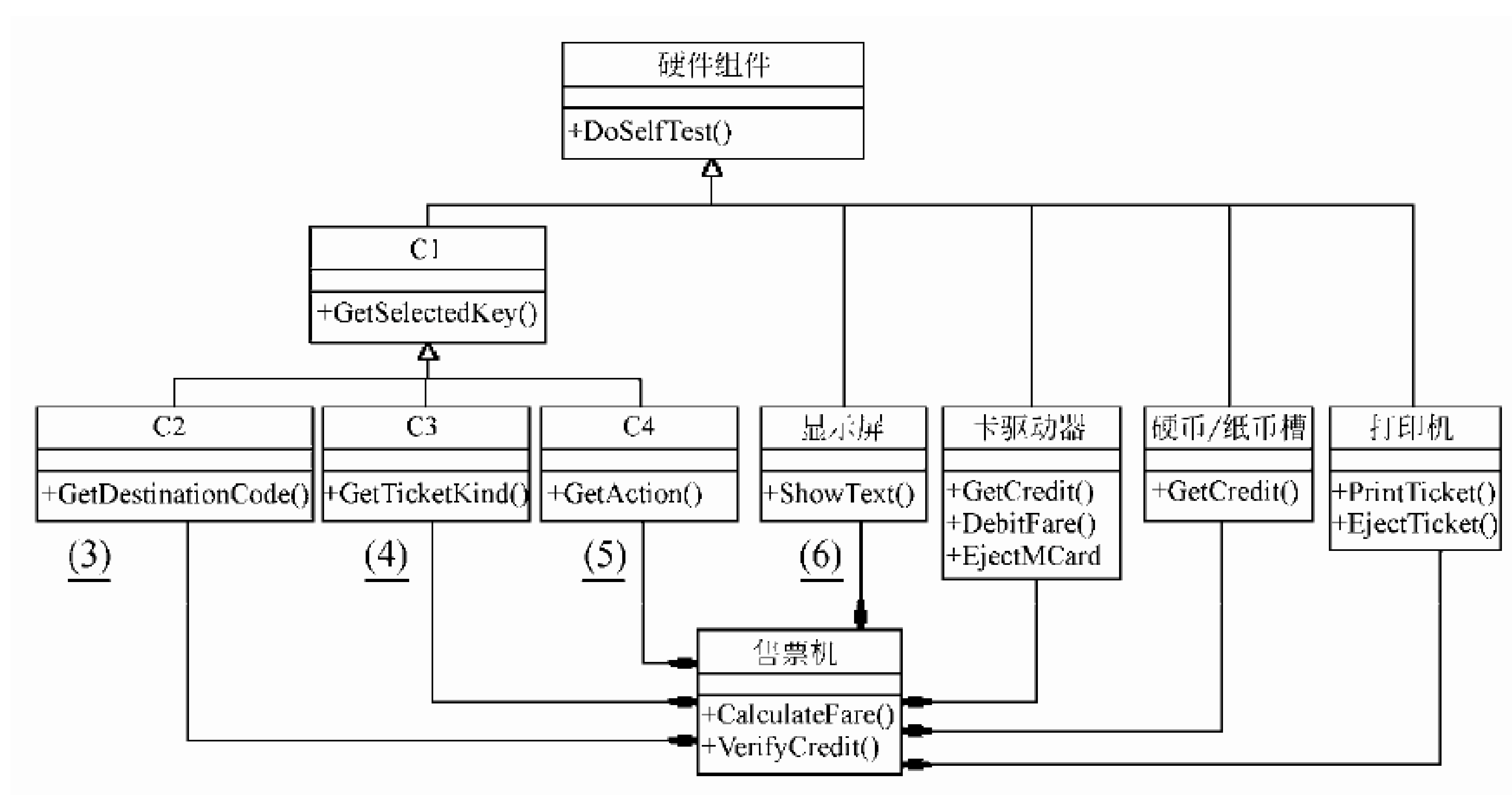


图 3-3 类图

【问题 1】（5 分）

根据说明中的描述，给出图 3-2 中 A1 和 A2 所对应的参与者，U1 所对应的用例，以及（1）、（2）处所对应的关系。

【问题 2】（7 分）

根据说明中的描述，给出图 3-3 中缺少的 C1～C4 所对应的类名以及（3）～（6）

处所对应的多重度。

【问题 3】（3 分）

图 3-3 中的类图设计采用了中介者（Mediator）设计模式，请说明该模式的内涵。

试题三分析

本题考查面向对象开发相关知识，涉及 UML 用例图、类图以及类图设计时的设计模式。UML 目前在面向对象软件开发中广泛使用，是面向对象软件开发考查的重要内容。

【问题 1】

本问题考查用例图。用例图用于确定系统边界，识别与系统交互的参与者，通过判断参与者发起的用例，建立和参与者之间的关联，然后再确认用例之间的关系。

本题中对售票机的描述为“乘客可以通过车票键盘选择车票种类（单程票、多次往返票和座席种类)；售票机的维护工作（取回现金、放入空白车票等）由服务技术人员完成”。由此可知，图 3-1 中 A1 为乘客，A2 为服务技术人员。

对购票用例，要选择目的地和车票类型、通过插卡口进行支付才可完成购票。因此 U2 为支付。

在考查用例之间的关系时，购票过程可以取消，也允许乘客连续购买多张票，因此，购票时可以包含多次选择目的地和车票类型、支付，即购票用例包含（关系<<include>>）选择目的地和车票类型以及支付。

【问题 2】

本问题考查类图。类图设计的重点是类的抽象和继承关系以及多重度。售票机的面板由多个控制部件组成。根据说明这些控制部件有目的地键盘、车票键盘和继续/取消键盘、显示屏、卡驱动器、硬币/纸币槽、打印机。图 3-3 中只有前 3 个部件在图中没有给出，而要填如 4 个类。从图中已经抽象出的硬件组件，给出了抽象的思路，从而可以把键盘抽象出来。由 C1 与 C2、C3、C4 的继承关系中 C1 为基类，可知 C1 为键盘。由 C2、C3 和 C4 给出的方法名称可知，C2 为目的地键盘获取目的地代码，C3 为车票键盘选择产品类型，C4 为继续/和取消动作。

本题中的重复度比较简单。从图 3-1 售票机的图示中可以看出，一个售票机只包含一个目的地键盘、一个车票键盘和一个继续/取消键盘，因此（3）～（6）均为 1。

【问题 3】

本问题考查设计模式。设计模式题目虽然比较难，但是本题题目中已经给出了所采用的设计模式为 Mediator 模式，只需说明设计模式的内涵即可，也比较容易。使用 Mediator 模式，可以使各个对象间的耦合松散，只需关心和 Mediator 的关系，使多对多的关系变成了一对多的关系，可以降低系统的复杂性，提高可修改扩展性。

参考答案

【问题 1】

A1：乘客　　　A2：服务技术人员

U1：支付　　(1) <<include>>　　(2) <<include>>

【问题2】

C1：键盘　　C2：目的地键盘　　C3：车票键盘　　C4：继续/取消键盘

(3) ～ (6)：1

【问题3】

使用Mediator模式，可以使各个对象间的耦合松散，只需关心和 Mediator的关系，使多对多的关系变成了一对多的关系，可以降低系统的复杂性，提高可修改扩展性。

试题四（共15分）

阅读下列说明和C代码，回答问题1至问题3，将解答写在答题纸的对应栏内。

【说明】

对有向图进行拓扑排序的方法是：

(1) 初始时拓扑序列为空；

(2) 任意选择一个入度为0的顶点，将其放入拓扑序列中，同时从图中删除该顶点以及从该顶点出发的弧；

(3) 重复 (2)，直到不存在入度为0的顶点为止（若所有顶点都进入拓扑序列则完成拓扑排序，否则由于有向图中存在回路无法完成拓扑排序）。

函数int* TopSort(LinkedDigraph G)的功能是对有向图G中的顶点进行拓扑排序，返回拓扑序列中的顶点编号序列，若不能完成拓扑排序，则返回空指针。其中，图G中的顶点从1开始依次编号，顶点序列为v1，v2，…，vn，图G采用邻接表表示，其数据类型定义如下：

```
#define MAXVNUM 50                      /* 最大顶点数 */
typedef struct ArcNode{                 /* 表结点类型 */
      int adjvex;                       /* 邻接顶点编号 */
      struct ArcNode *nextarc;          /* 指示下一个邻接顶点 */
}ArcNode;
typedef struct AdjList{                 /* 头结点类型 */
      char vdata;                       /* 顶点的数据信息 */
      ArcNode *firstarc;                /* 指向邻接表的第一个表结点 */
}AdjList;
typedef struct LinkedDigraph{           /* 图的类型 */
      int n;                            /* 图中顶点个数 */
      AdjList Vhead[MAXVNUM];           /* 所有顶点的头结点数组 */
}LinkedDigraph;
```

例如，某有向图G如图4-1所示，其邻接表如图4-2所示。

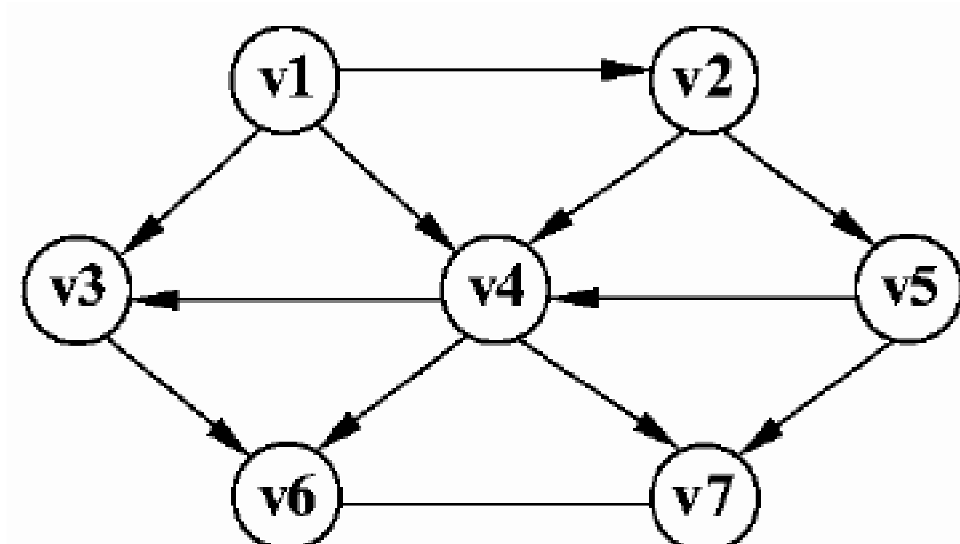


图 4-1　有向图 G

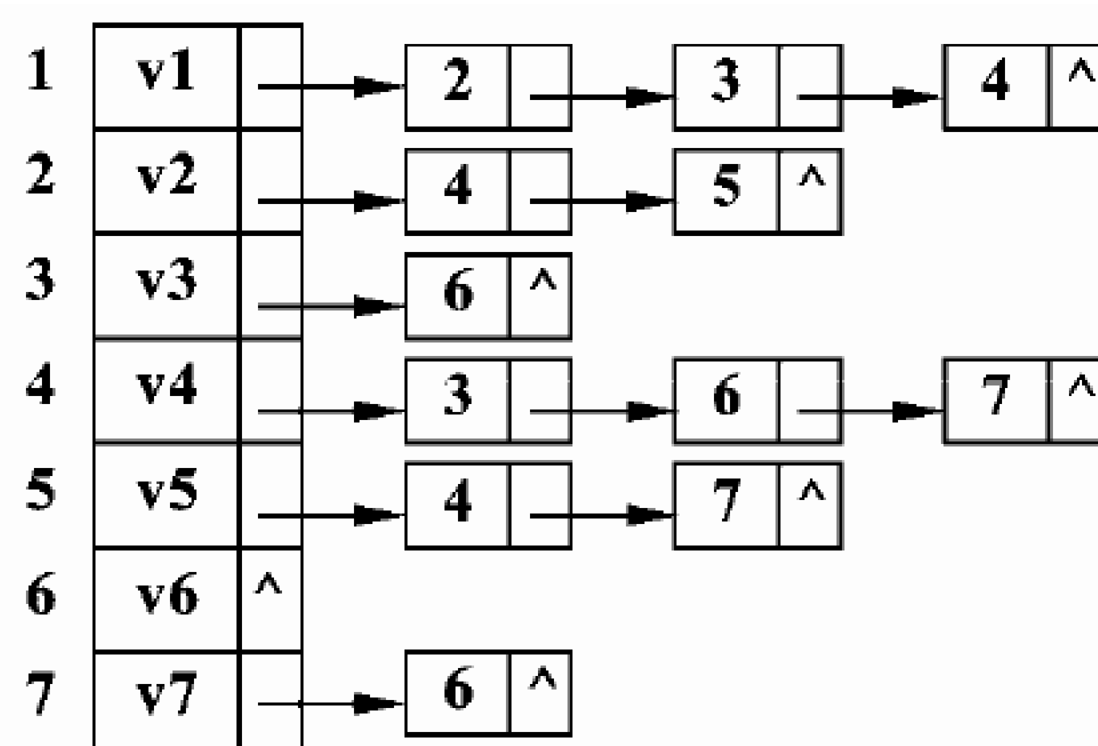


图 4-2　有向图 G 的邻接表示意图

函数 TopSort 中用到了队列结构（Queue 的定义省略），实现队列基本操作的函数原型如下表所示：

函 数 原 型	说　　明
void InitQueue(Queue *Q)	初始化队列（构造一个空队列）
bool IsEmpty(Queue Q)	判断队列是否为空，若是则返回 true，否则返回 false
void EnQueue(Queue *Q, int e)	元素入队列
void DeQueue(Queue *Q,int *p)	元素出队列

【C 代码】

```
int *TopSort(LinkedDigraph G) {
    ArcNode *p;                    /* 临时指针，指示表结点 */
Queue Q;                           /* 临时队列，保存入度为 0 的顶点编号 */
int k = 0;                         /* 临时变量，用作数组元素的下标 */
int j = 0, w = 0;                  /* 临时变量，用作顶点编号 */
int  *topOrder, *inDegree;
topOrder = (int *)malloc((G.n+1) * sizeof(int));
                                   /* 存储拓扑序列中的顶点编号 */
inDegree = (int *)malloc((G.n+1) * sizeof(int));
                                   /* 存储图 G 中各顶点的入度 */
if (!inDegree || !topOrder) return NULL;

____(1)____;                       /* 构造一个空队列 */

for ( j = 1; j <= G.n; j++ ) {  /* 初始化 */
     topOrder[j] = 0;    inDegree[j] = 0;
}
```

```
    for (j = 1; j <= G.n; j++)  /* 求图 G 中各顶点的入度 */
    for( p = G.Vhead[j].firstarc; p; p = p->nextarc )
            inDegree[p-> adjvex] += 1;

for (j = 1; j <= G.n; j++)       /* 将图 G 中入度为 0 的顶点保存在队列中 */
if ( 0 == inDegree[j] )     EnQueue(&Q,j);

while (!IsEmpty(Q)) {
          ___(2)___;                /* 队头顶点出队列并用 w 保存该顶点的编号 */
topOrder[k++] = w;
/* 将顶点 w 的所有邻接顶点的入度减 1（模拟删除顶点 w 及从该顶点出发的弧的操作） */
          for(p = G.Vhead[w].firstarc; p; p = p->nextarc) {
              ___(3)___ -= 1;
              if (0 ==___(4)___)    EnQueue(&Q, p->adjvex);
     }/* for */
     }/* while */

    free(inDegree);

    if (___(5)___)
          return NULL;
return topOrder;
} /*TopSort*/
```

【问题 1】（9 分）

根据以上说明和 C 代码，填充 C 代码中的空（1）～（5）。

【问题 2】（2 分）

对于图 4-1 所示的有向图 G，写出函数 TopSort 执行后得到的拓扑序列。若将函数 TopSort 中的队列改为栈，写出函数 TopSort 执行后得到的拓扑序列。

【问题 3】（4 分）

设某有向无环图的顶点个数为 n、弧数为 e，那么用邻接表存储该图时，实现上述拓扑排序算法的函数 TopSort 的时间复杂度是___(6)___。

若有向图采用邻接矩阵表示（例如，图 4-1 所示有向图的邻接矩阵如图 4-3 所示），且将函数 TopSort 中有关邻接表的操作修改为针对邻接矩阵的操作，那么对于有 n 个顶点、e 条弧的有向无环图，实现上述拓扑排序算法的时间复杂度是___(7)___。

	v1	v2	v3	v4	v5	v6	v7
v1	0	1	1	1	0	0	0
v2	0	0	0	1	1	0	0
v3	0	0	0	0	0	1	0
v4	0	0	1	0	0	1	1
v5	0	0	0	1	0	0	1
v6	0	0	0	0	0	0	0
v7	0	0	0	0	0	1	0

图 4-3　有向图 G 的邻接矩阵

试题四分析

本题考查数据结构和算法中的拓扑排序算法。

【问题 1】

拓扑排序是将有向无环图中所有顶点排成一个线性序列的过程，并且该序列满足：若在有向图中从顶点 v_i 到 v_j 有一条路径，则在该线性序列中，顶点 v_i 必然在顶点 v_j 之前。

对 AOE 网进行拓扑排序的方法如下：

① 在 AOE 网中选择一个入度为零（没有前驱）的顶点且输出它；

② 从网中删除该顶点及其与该顶点有关的所有边；

③ 重复上述两步，直至网中不存在入度为零的顶点为止。

在拓扑排序过程中，需要将入度为 0 的顶点临时存储起来。函数中用一个队列暂存入度为 0 且没有进入拓扑序列的顶点。显然，空（1）处应填入 InitQueue(&Q)。

进行拓扑排序之前，应先求出网中每个顶点的入度并存入数组 inDegree[]中，从而将“从网中删除该顶点及其与该顶点有关的所有边”的操作转换为“相关顶点的入度减 1”，一旦发现某个顶点的入度变为 0，就将其编号压入堆栈。从而将选择入度为 0 的顶点操作转化为令队头所代表的顶点出队。

根据注释，空（2）处应填入 DeQueue(&Q,&w)，实现队头元素出队列的处理。

题中图采用邻接表存储结构，当指针 p 指向 v_i 邻接表中的结点时，p->adjvex 表示 v_i 的一个邻接顶点，删除 v_i 至顶点 p-> adjvex 的弧的操作实现为顶点 p->adjvex 的入度减 1，因此，空（3）处应填入 inDegree[p->adjvex]，当顶点 p->adjvex 的入度为 0 时，需要将其加入队列，因此空（4）处也应填入 inDegree[p->adjvex]。

空（5）处判断是否所有顶点都加入了拓扑序列，算法中变量 k 用于对加入序列的顶点计数，因此，空（5）处应填入“k < G.n”或“k !=G.n”。

【问题 2】

使用栈和队列的差别在于拓扑序列中顶点的排列次序可能不同。对于本题中的有向图，在使用队列的方式下：

（1）开始时仅顶点 v1 的入度为 0，因此顶点 v1 入队；

（2）队头顶点 v1 出队，并进入拓扑序列，然后删除从顶点 v1 出发的弧后，仅使顶点 v2 的入度为 0，因此顶点 v2 入队；

（3）队头顶点 v2 出队，并进入拓扑序列，然后删除从顶点 v2 出发的弧后，仅使顶点 v5 的入度为 0，因此顶点 v5 入队；

（4）队头顶点 v5 出队，并进入拓扑序列，然后删除从顶点 v5 出发的弧后，仅使顶点 v4 的入度为 0，因此顶点 v4 入队；

（5）队头顶点 v4 出队，并进入拓扑序列，然后删除从顶点 v4 出发的弧后，仅使顶点 v3 和 v7 的入度为 0，因此顶点 v3 和 v7 依次入队；

（6）队头顶点 v3 出队，并进入拓扑序列，然后删除从顶点 v3 出发的弧后，没有产生新的入度为 0 的顶点；

（7）队头顶点 v7 出队，并进入拓扑序列，然后删除从顶点 v7 出发的弧后，使顶点 v6 的入度为 0，因此顶点 v6 入队；

（8）队头顶点 v6 出队，并进入拓扑序列，然后删除从顶点 v6 出发的弧后，没有产生新的入度为 0 的顶点，队列已空，因此结束拓扑排序过程，得到的拓扑序列为 v1 v2 v5 v4 v3 v7 v6。

使用栈保存入度为 0 的顶点时，前 4 步都是一样的，因为每次仅有一个元素进栈，因此出栈序列与入栈序列一致。到第 5 步时，v3 和 v7 依次入栈后，出栈时的次序为 v7 和 v3，因此得到的拓扑序列为 v1 v2 v5 v4 v7 v3 v6。

【问题 3】

以邻接表为存储结构时，计算各顶点入度的时间复杂度为 $O(e)$，建立零入度顶点队列的时间复杂度为 $O(n)$。在拓扑排序过程中，（图中无环的情况下）每个顶点进出队列各 1 次，入度减 1 的操作在 while 循环中共执行 e 次，所以总的时间复杂度为 $O(n+e)$。

以邻接矩阵为存储结构时，计算各顶点入度时需要遍历整个矩阵，因此时间复杂度为 $O(n^2)$，建立零入度顶点队列的时间复杂度为 $O(n)$。在拓扑排序过程中，（图中无环的情况下）每个顶点进出队列各 1 次，实现入度减 1 操作时需遍历每个顶点的行向量 1 遍（时间复杂度为 $O(n)$），所以总的时间复杂度为 $O(n^2)$。

参考答案

【问题 1】

（1）InitQueue(&Q)

（2）DeQueue(&Q,&w)

（3）inDegree[p-> adjvex]　　或其等价形式

（4）inDegree[p->adjvex]　　或其等价形式

（5）k < G.n　　或 k !=G.n　　或其等价形式

【问题 2】

队列方式：v1 v2 v5 v4 v3 v7 v6　　或者 1 2 5 4 3 7 6

栈方式：v1 v2 v5 v4 v7 v3 v6　　或者 1 2 5 4 7 3 6

【问题 3】

（6）$O(n+e)$　（7）$O(n^2)$

试题五（共 15 分）

阅读下列说明和 C++代码，将应填入 __（n）__ 处的字句写在答题纸的对应栏内。

【说明】

某软件公司现欲开发一款飞机飞行模拟系统，该系统主要模拟不同种类飞机的飞行特征与起飞特征。需要模拟的飞机种类及其特征如表 5-1 所示。

表 5-1

飞 机 种 类	起 飞 特 征	飞 行 特 征
直升机（Helicopter）	垂直起飞（VerticalTakeOff）	亚音速飞行（SubSonicFly）
客机（AirPlane）	长距离起飞（LongDistanceTakeOff）	亚音速飞行（SubSonicFly）
歼击机（Fighter）	长距离起飞（LongDistanceTakeOff）	超音速飞行（SuperSonicFly）
鹞式战斗机（Harrier）	垂直起飞（VerticalTakeOff）	超音速飞行（SuperSonicFly）

为支持将来模拟更多种类的飞机，采用策略设计模式（Strategy）设计的类图如图 5-1 所示。

图 5-1 中，AirCraft 为抽象类，描述了抽象的飞机，而类 Helicopter、AirPlane、Fighter 和 Harrier 分别描述具体的飞机种类，方法 fly()和 takeOff()分别表示不同飞机都具有飞行特征和起飞特征；类 FlyBehavior 与 TakeOffBehavior 为抽象类，分别用于表示抽象的飞行行为与起飞行为；类 SubSonicFly 与 SuperSonicFly 分别描述亚音速飞行和超音速飞行的行为；类 VerticalTakeOff 与 LongDistanceTakeOff 分别描述垂直起飞与长距离起飞的行为。

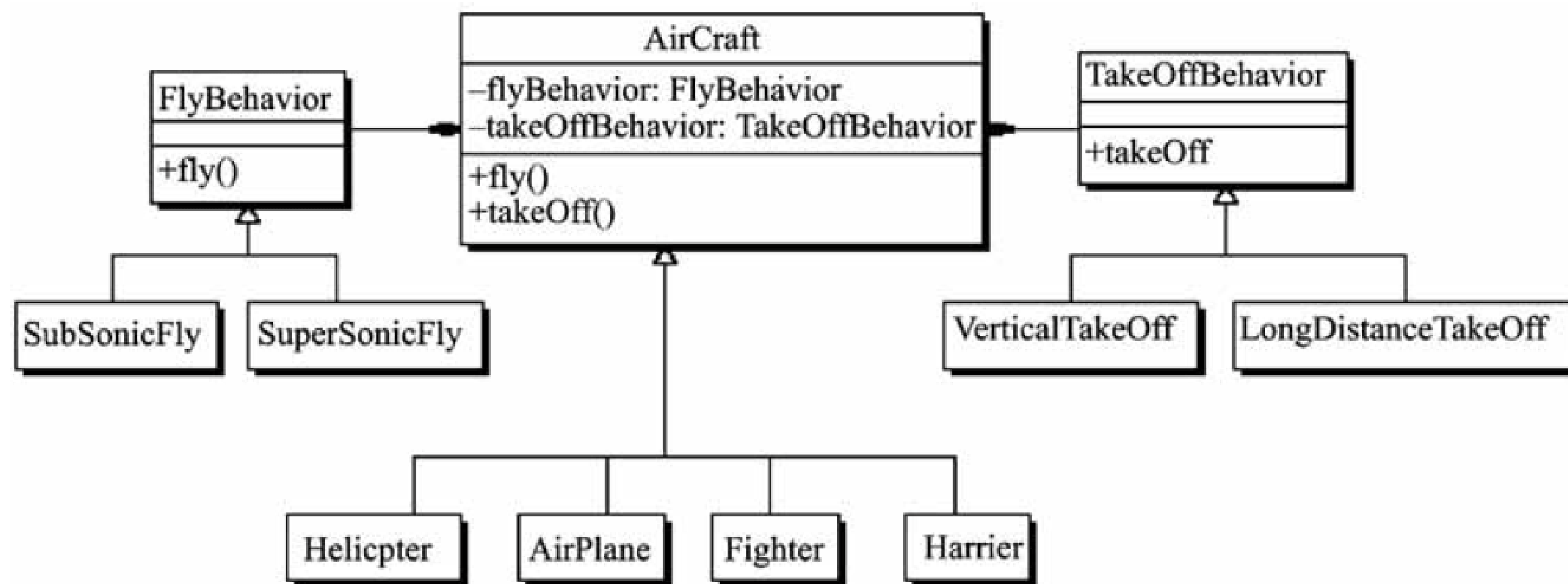


图5-1 类图

【C++代码】

```
#include<iostream>
using namespace std;
class FlyBehavior {
public : virtual void fly() = 0;
};
class SubSonicFly:public FlyBehavior{
public: void fly(){ cout << "亚音速飞行 ! " << endl; }
};
class SuperSonicFly:public FlyBehavior{
public: void fly(){ cout << "超音速飞行 ! " << endl; }
};

class TakeOffBehavior {
public: virtual void takeOff() = 0;
};
class VerticalTakeOff:public TakeOffBehavior{
public: void takeOff(){ cout << "垂直起飞 ! " << endl;    }
};
class LongDistanceTakeOff:public TakeOffBehavior {
public: void takeOff (){ cout << "长距离起飞 ! " << endl; }
};

class AirCraft{
protected:
```

```
    (1)   ;
     (2)   ;
public:
    void fly(){   (3)   ;  }
    void takeOff() {   (4)   ; };
};
class Helicopter: public AirCraft {
public:
    Helicopter (){
        flyBehavior = new   (5)   ;
        takeOffBehavior = new   (6)   ;
    }
      (7)   {
        if(!flyBehavior) delete flyBehavior;
        if(!takeOffBehavior) delete takeOffBehavior;
    }
};
//其他代码省略
```

试题五分析

本题考查了设计模式中的策略设计模式，实际上与 2007 年上半年考核内容相同。

从本题的叙述中可以看出，存在 4 种不同的飞机类型，但每种飞机类型的起飞特征和飞行特征并不完全相同，这就使得我们很难采用比较直接的方法来实现重用。例如，定义一个抽象的飞机类，实现飞机的起飞特征，然后 4 种飞机直接重用该特征。但是，我们可以观察到，尽管飞机的起飞特征和飞行特征有所不同，有一点可以肯定的是，每一种飞机都具备了飞行特征和起飞特征。因此，可以抽象出一个飞机类，其中含有飞行特征与起飞特征，但关于两个特征的实现要单独抽取出来，所以又形成了 FlyBehavior 类和 TakeOffBehavior 类分别表示抽象的飞行和起飞特征，而这两个类的子类则分别实现不同的起飞和飞行特征，最终转化为，在创建一个具体的飞机时，给其配上不同的起飞特征和飞行特征即可。

本题中的空（1）和空（2）应该填写成员变量，根据类图可以得知，此处应该表示的是飞行和起飞特征变量，在 C++中可以采用指针来表示。空（3）和空（4）处需要实现飞行与起飞特征，但 AirCraft 是抽象的类，所以把实现代理给指针变量。Helicopter 类需要指定由父类继承而来的成员变量的初始值，因为 Helicopter 的特征是垂直起飞和亚音速飞行，因此生成这两个特征的对象，分别赋值给 flyBehavior 和 takeOffBehavior

变量。

参考答案

（1）FlyBehavior * flyBehavior

（2）TakeOffBehavior * takeOffBehavior

（3）flyBehavior->fly()

（4）takeOffBehavior->takeOff()

（5）SubSonicFly()

（6）VerticalTakeOff()

（7）~ Helicopter()

注：空（1）与空（2）答案可互换

试题六（共 15 分）

阅读下列说明和 Java 代码，将应填入 （n） 处的字句写在答题纸的对应栏内。

【说明】

某软件公司现欲开发一款飞机飞行模拟系统，该系统主要模拟不同种类飞机的飞行特征与起飞特征。需要模拟的飞机种类及其特征如表 6-1 所示。

表 6-1

飞 机 种 类	起 飞 特 征	飞 行 特 征
直升机（Helicopter）	垂直起飞（VerticalTakeOff）	亚音速飞行（SubSonicFly）
客机（AirPlane）	长距离起飞（LongDistanceTakeOff）	亚音速飞行（SubSonicFly）
歼击机（Fighter）	长距离起飞（LongDistanceTakeOff）	超音速飞行（SuperSonicFly）
鹞式战斗机（Harrier）	垂直起飞（VerticalTakeOff）	超音速飞行（SuperSonicFly）

为支持将来模拟更多种类的飞机，采用策略设计模式（Strategy）设计的类图如图 6-1 所示。

图 6-1 中，AirCraft 为抽象类，描述了抽象的飞机，而类 Helicopter、AirPlane、Fighter 和 Harrier 分别描述具体的飞机种类，方法 fly()和 takeOff()分别表示不同飞机都具有飞行特征和起飞特征；类 FlyBehavior 与 TakeOffBehavior 为抽象类，分别用于表示抽象的飞行行为与起飞行为；类 SubSonicFly 与 SuperSonicFly 分别描述亚音速飞行和超音速飞行的行为；类 VerticalTakeOff 与 LongDistanceTakeOff 分别描述垂直起飞与长距离起飞的行为。

【Java 代码】

```
interface FlyBehavior {
    public void fly();
};
class SubSonicFly implements FlyBehavior{
    public void fly(){ System.out.println("亚音速飞行 ！"); }
};
class SuperSonicFly implements FlyBehavior{
    public void fly(){ System.out.println("超音速飞行 ！" ); }
};
```

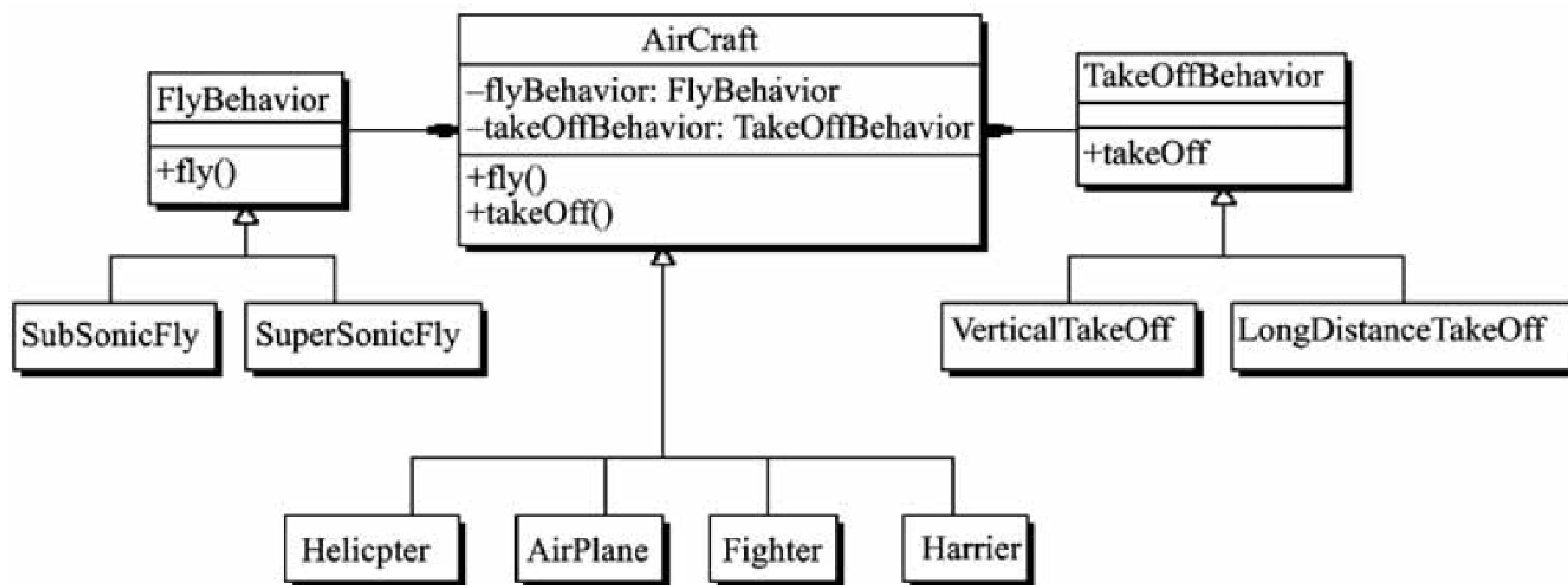


图 6-1　类图

```
interface TakeOffBehavior {
    public void takeOff();
};

class VerticalTakeOff implements TakeOffBehavior {
    public void takeOff (){ System.out.println("垂直起飞 ！" );   }
};
class LongDistanceTakeOff implements TakeOffBehavior {
    public void takeOff(){ System.out.println("长距离起飞 ！");   }
};

abstract class AirCraft {
    protected   (1)  ;
    protected   (2)  ;
    public void fly(){   (3)  ;  }
    public void takeOff() {   (4)  ; };
```

```
};
class Helicopter ___(5)___ AirCraft{
    public Helicopter (){
        flyBehavior = new ___(6)___;
        takeOffBehavior = new ___(7)___;
    }
};
//其他代码省略
```

试题六分析

本题考查了设计模式中的策略设计模式，实际上与 2007 年上半年 Java 题目的考核内容相同。

从本题的叙述中可以看出，存在四种不同的飞机类型，但每种飞机类型的起飞特征和飞行特征并不完全相同，这就使得我们很难采用比较直接的方法来实现重用。例如，定义一个抽象的飞机类，实现飞机的起飞特征，然后四种飞机直接重用该特征。但是，我们可以观察到，尽管飞机的起飞特征和飞行特征有所不同，有一点可以肯定的是，每一种飞机都具备了飞行特征和起飞特征。因此，可以抽象出一个飞机类，其中含有飞行特征与起飞特征，但关于两个特征的实现要单独抽取出来，所以又形成了 FlyBehavior 类和 TakeOffBehavior 类，分别表示抽象的飞行和起飞特征，而这两个类的子类则分别实现不同的起飞和飞行特征，最终转化为，在创建一个具体的飞机时，给其配上不同的起飞特征和飞行特征即可。

本题中的空（1）和空（2）应该填写成员变量，根据类图可以得知，此处应该表示的是飞行和起飞特征变量。空（3）和空（4）处需要实现飞行与起飞特征，但 AirCraft 是抽象的类，所以把实现代理给指针变量。Helicopter 类需要指定由父类继承而来的成员变量的初始值，因为 Helicopter 的特征是垂直起飞和亚音速飞行，因此生成这两个特征的对象，分别赋值给 flyBehavior 和 takeOffBehavior 变量。

参考答案

（1）FlyBehavior flyBehavior

（2）TakeOffBehavior takeOffBehavior

（3）flyBehavior.fly()

（4）takeOffBehavior.takeOff()

（5）extends

（6）SubSonicFly()

（7）VerticalTakeOff()

第 7 章　2010 下半年软件设计师上午试题分析与解答

试题（1）

在输入输出控制方法中，采用＿（1）＿可以使得设备与主存间的数据块传送无须CPU 干预。

（1）A. 程序控制输入输出　　B. 中断

C. DMA　　D. 总线控制

试题（1）分析

本题考查 CPU 中相关寄存器的基础知识。

计算机中主机与外设间进行数据传输的输入输出控制方法有程序控制方式、中断方式、DMA 等。

在程序控制方式下，由 CPU 执行程序控制数据的输入输出过程。

在中断方式下，外设准备好输入数据或接收数据时向 CPU 发出中断请求信号，若CPU 决定响应该请求，则暂停正在执行的任务，转而执行中断服务程序进行数据的输入输出处理，之后再回去执行原来被中断的任务。

在 DMA 方式下，CPU 只需向 DMA 控制器下达指令，让 DMA 控制器来处理数据的传送，数据传送完毕再把信息反馈给 CPU，这样就很大程度上减轻了 CPU 的负担，可以大大节省系统资源。

参考答案

（1）C

试题（2）

若某计算机采用 8 位整数补码表示数据，则运算＿（2）＿将产生溢出。

（2）A. –127+1　　B. –127–1　　C. 127+1　　D. 127–1

试题（2）分析

本题考查计算机中的数据表示和运算基础知识。

采用 8 位补码表示整型数据时，可表示的数据范围为–128～127，因此进行 127+1运算会产生溢出。

参考答案

（2）C

试题（3）

若内存容量为 4GB，字长为 32，则＿（3）＿。

（3）A. 地址总线和数据总线的宽度都为 32

B. 地址总线的宽度为 30，数据总线的宽度为 32

C. 地址总线的宽度为 30，数据总线的宽度为 8

D. 地址总线的宽度为 32，数据总线的宽度为 8

试题（3）分析

本题考查计算机系统的总线基础知识。

内存容量为 4GB，即内存单元的地址宽度为 32 位。字长为 32 位即要求数据总线的宽度为 32 位，因此地址总线和数据总线的宽度都为 32。

参考答案

（3）A

试题（4）

设用 2K×4 位的存储器芯片组成 16K×8 位的存储器（地址单元为 0000H～3FFFH，每个芯片的地址空间连续），则地址单元 0B1FH 所在芯片的最小地址编号为＿（4）＿。

（4）A. 0000H　　B. 0800H　　C. 2000H　　D. 2800

试题（4）分析

本题考查计算机系统中存储部件的基础知识。

由 2K×4 位的存储器芯片组成容量为 16K×8 位的存储器时，共需要 16 片（16K×8/（2K×4））。用 2 个存储器芯片组成 2K×8 的存储空间（每个芯片的地址空间连续），16K×8 位的存储空间共分为 8 段，即 0000H～07FFH，0800H～0FFFH，1000H～17FFH，1800H～1FFFH，2000H～27FFH，2800H～2FFFH，3000H～37FFH，3800H～3FFFH。显然，地址单元 0B1FH 所在芯片的起始地址为 0800H。

参考答案

（4）B

试题（5）

编写汇编语言程序时，下列寄存器中程序员可访问的是＿（5）＿。

（5）A. 程序计数器（PC）　　B. 指令寄存器（IR）

C. 存储器数据寄存器（MDR）　　D. 存储器地址寄存器（MAR）

试题（5）分析

本题考查 CPU 中相关寄存器的基础知识。

指令寄存器（IR）用于暂存从内存取出的、正在运行的指令，这是由系统使用的寄存器，程序员不能访问。

存储器数据寄存器（MDR）和存储器地址寄存器（MAR）用于对内存单元访问时的数据和地址暂存，也是由系统使用的，程序员不能访问。

程序计数器（PC）用于存储指令的地址，CPU 根据该寄存器的内容从内存读取待执行的指令，程序员可以访问该寄存器。

参考答案

（5）A

试题（6）

正常情况下，操作系统对保存有大量有用数据的硬盘进行__（6）__操作时，不会清除有用数据。

（6）A. 磁盘分区和格式化　　B. 磁盘格式化和碎片整理
C. 磁盘清理和碎片整理　　D. 磁盘分区和磁盘清理

试题（6）分析

本题考查计算机系统的基础知识。

磁盘格式化是指把一张空白的盘划分成一个个小区域并编号，以供计算机储存和读取数据。格式化是一种纯物理操作，是在磁盘的所有数据区上写零的操作过程，同时对硬盘介质做一致性检测，并且标记出不可读和坏的扇区。由于大部分硬盘在出厂时已经格式化过，所以只有在硬盘介质产生错误时才需要进行格式化。

磁盘分区就是将磁盘划分成一块块的存储区域。在传统的磁盘管理中，将一个硬盘分为两大类分区：主分区和扩展分区。主分区是能够安装操作系统、能够进行计算机启动的分区，这样的分区可以直接格式化，然后安装系统，直接存放文件。

磁盘里的文件都是按存储时间先后来排列的，理论上文件之间都是紧凑排列而没有空隙的。但是，用户常常会对文件进行修改，而且新增加的内容并不是直接加到原文件的位置的，而是放在磁盘存储空间的最末尾，系统会在这两段之间加上联系标识。当有多个文件被修改后，磁盘里就会有很多不连续的文件。一旦文件被删除，所占用的不连续空间就会空着，并不会被自动填满，而且，新保存的文件也不会放在这些地方，这些空着的磁盘空间，就被称作"磁盘碎片"。因此，硬盘的每个分区里都会有碎片。碎片太多，其他的不连续文件相应也多，系统在执行文件操作时就会因反复寻找联系标识，工作效率大大降低，直接的反映就是感觉慢。

磁盘清理将删除计算机上所有不需要的文件（这些文件由用户或系统进行确认）。

磁盘碎片整理，就是通过系统软件或者专业的磁盘碎片整理软件对电脑磁盘在长期使用过程中产生的碎片和凌乱文件重新整理，释放出更多的磁盘空间，可提高电脑的整体性能和运行速度。

参考答案

（6）C

试题（7）

如果使用大量的连接请求攻击计算机，使得所有可用的系统资源都被消耗殆尽，最终计算机无法再处理合法用户的请求，这种手段属于__（7）__攻击。

（7）A. 拒绝服务　　B. 口令入侵　　C. 网络监听　　D. IP 欺骗

试题（7）分析

本题考查网络安全中网络攻击的基础知识。

网络攻击的主要手段包括口令入侵、放置特洛伊木马程序、拒绝服务（DoS）攻击、端口扫描、网络监听、欺骗攻击和电子邮件攻击等。

口令入侵是指使用某些合法用户的账号和口令登录到目的主机，然后再实施攻击活动。

特洛伊木马（Trojans）程序常被伪装成工具程序或游戏，一旦用户打开了带有特洛伊木马程序的邮件附件或从网上直接下载，或执行了这些程序之后，当用户连接到互联网上时，这个程序就会向黑客通知用户的 IP 地址及被预先设定的端口。

拒绝服务（DoS）攻击目的是使计算机或网络无法提供正常的服务。最常见的拒绝服务攻击有网络带宽攻击和连通性攻击。带宽攻击指以极大的通信量冲击网络，使得所有可用网络资源都被消耗殆尽，最后导致合法的用户请求无法通过。连通性攻击是指用大量的连接请求冲击计算机，使得所有可用的操作系统资源都被消耗殆尽，最终计算机无法再处理合法用户的请求。

端口扫描就是利用 Socket 编程与目标主机的某些端口建立 TCP 连接、进行传输协议的验证等，从而侦知目标主机的扫描端口是否处于激活状态、主机提供了哪些服务、提供的服务中是否含有某些缺陷等。

网络监听是主机的一种工作模式，在这种模式下，主机可以接收到本网段在同一条物理通道上传输的所有信息。使用网络监听工具可轻而易举地截取包括口令和账号在内的信息资料。

欺骗攻击是攻击者创造一个易于误解的上下文环境，以诱使受攻击者进入并且做出缺乏安全考虑的决策。IP 欺骗是欺骗攻击的一种，IP 欺骗实现的过程是：使得被信任的主机丧失工作能力，同时采样目标主机发出的 TCP 序列号，猜测出它的数据序列号。然后，伪装成被信任的主机，同时建立起与目标主机基于地址验证的应用连接。如果成功，黑客可以使用一种简单的命令放置一个系统后门，以进行非授权操作。

参考答案

（7）A

试题（8）

ARP 攻击造成网络无法跨网段通信的原因是__（8）__。

（8）A. 发送大量 ARP 报文造成网络拥塞

B. 伪造网关 ARP 报文使得数据包无法发送到网关

C. ARP 攻击破坏了网络的物理连通性

D. ARP 攻击破坏了网关设备

试题（8）分析

本题考查网络攻击中 ARP 攻击的原理。

ARP 攻击（ARP 欺骗）是欺骗攻击的一种，通过伪造 IP 地址和 MAC 地址，能够在网络中产生大量的 ARP 通信量使网络阻塞，如果伪造网关的 IP 地址和 MAC 地址对，则所有发往网关的 IP 包将因为 MAC 地址错误而无法到达网关(ARP 攻击一般会将 MAC 地址改为发起 ARP 攻击的主机地址)，造成无法跨网段通信。

处理 ARP 攻击的方法为首先断开 ARP 攻击主机的网络连接，然后用“arp-d”命令清除受攻击影响的 ARP 缓存。

参考答案

（8）B

试题（9）

下列选项中，防范网络监听最有效的方法是__（9）__。

（9）A. 安装防火墙　　B. 采用无线网络传输
　　C. 数据加密　　D. 漏洞扫描

试题（9）分析

本题考查网络攻击中网络监听的基础知识。

网络监听是主机的一种工作模式，在这种模式下，主机可以接收到本网段在同一条物理通道上传输的所有信息。使用网络监听工具可轻而易举地截取包括口令和账号在内的信息资料。采用数据加密的方式保护包括口令和账号在内的信息资料，使得即使网络监听获取密文后也无法解密成明文，是对付网络监听的有效手段。

参考答案

（9）C

试题（10）

软件商标权的权利人是指__（10）__。

（10）A. 软件商标设计人　　B. 软件商标制作人
　　C. 软件商标使用人　　D. 软件注册商标所有人

试题（10）分析

本题考查知识产权方面的基础知识，涉及软件商标权主体资格的相关概念。

在我国，商标权是指注册商标专用权，只有依法进行商标注册后，商标注册人才能取得商标权，其商标才能得到法律的保护。商标权不包括商标设计人的权利，主要注重商标所有人的权利，即注册商标所有人具有其商标的专用权。商标设计人的发表权、署名权等人身权在商标的使用中没有反映，所以不受商标法保护。商标设计人可以通过其他法律来保护属于自己的权利，如可以将商标设计图案作为美术作品通过著作权法来保护；与产品外观关系密切的商标图案还可以申请外观设计专利通过专利法加以保护。软件商标制作人、软件商标使用人均未涉及软件注册商标，所以均不能成为软件商标权的权利人。

参考答案

（10）D

试题（11）

利用__（11）__可以对软件的技术信息、经营信息提供保护。

（11）A. 著作权　B. 专利权　C. 商业秘密权　D. 商标权

试题（11）分析

本题考查知识产权方面的基础知识，涉及软件商业秘密权的相关概念。

著作权从软件作品性的角度保护其表现形式，源代码（程序）、目标代码（程序）、软件文档是计算机软件的基本表达方式（表现形式），受著作权保护；专利权从软件功能性的角度保护软件的思想内涵，即软件的技术构思、程序的逻辑和算法等的思想内涵，当计算机软件同硬件设备是一个整体，涉及计算机程序的发明专利，可以申请方法专利，取得专利权保护。商标权是为商业化的软件从商品、商誉的角度为软件提供保护，利用商标权可以禁止他人使用相同或者近似的商标、生产（制作）或销售假冒软件产品。商标权受保护的力度大于其他知识产权，对软件的侵权行为更容易受到行政查处。而商业秘密权是商业秘密的合法控制人采取了保密措施，依法对其经营信息和技术信息享有的专有使用权，我国《反不正当竞争法》中对商业秘密的定义为"不为公众所知悉、能为权利人带来经济利益、具有实用性并经权利人采取保密措施的技术信息和经营信息"。软件技术秘密是指软件中适用的技术情报、数据或知识等，包括程序、设计方法、技术方案、功能规划、开发情况、测试结果及使用方法的文字资料和图表，如程序设计说明书、流程图、用户手册等。软件经营秘密指具有软件秘密性质的经营管理方法以及与经营管理方法密切相关的信息和情报，其中包括管理方法、经营方法、产销策略、客户情报（客户名单、客户需求），以及对软件市场的分析、预测报告和未来的发展规划、招投标中的标底及标书内容等。

参考答案

（11）C

试题（12）

李某在某软件公司兼职，为完成该公司交给的工作，做出了一项涉及计算机程序的发明。李某认为该发明是自己利用业余时间完成的，可以个人名义申请专利。关于此项发明的专利申请权应归属__（12）__。

（12）A. 李某　B. 李某所在单位

C. 李某兼职的软件公司　D. 李某和软件公司约定的一方

试题（12）分析

本题考查知识产权方面的基础知识，涉及软件发明专利申请权归属的相关概念。

根据《专利法》第六条第1款规定，执行本单位的任务所完成的发明创造是职务发明创造。职务发明创造申请专利的权利属于单位，申请被批准后，该单位为专利权人。

《专利法实施细则》第十一条对“执行本单位的任务所完成的发明创造”作出了解释。执行本单位的任务所完成的发明创造是指：（1）在本职工作中作出的发明创造；（2）履行本单位交付的本职工作之外的任务所作出的发明创造；（3）退职、退休或者调动工作后一年内所作出的、与其在原单位承担的本职工作或原单位分配的任务有关的发明创造。李某是为完成其兼职软件公司交给的工作而作出的该项发明，属于职务发明。专利申请权应归属软件公司。

《专利法》第六条第 3 款规定：“利用本单位的物质技术条件所完成的发明创造，单位与发明人或者设计人订有合同，对申请专利的权利和专利权的归属作出约定的，从其约定。”在事先有约定的情况下，按照约定确定权属。如果单位和发明人没有对权属问题作出约定或约定不明的，该发明创造仍视为职务发明创造，专利申请权仍然属于单位。本题未涉及合同约定，故 D 项不正确。

参考答案

（12）C

试题（13）

一幅彩色图像（RGB），分辨率为 256×512，每一种颜色用 8b 表示，则该彩色图像的数据量为<u>（13）</u>b。

（13）A. 256×512×8　　B. 256×512×3×8

C. 256×512×3/8　　D. 256×512×3

试题（13）分析

本题考查多媒体方面的基础知识，涉及彩色图像数据量计算。

图像的分辨率越高，图像深度越深，则数字化后的图像效果越逼真，图像数据量也越大。其图像数据量可用下面的公式估算：

图像数据量=图像的总像素×图像深度　（b）

其中图像的总像素为图像的水平方向像素乘以垂直方向像素数。例如，一幅 640×480 的 256 色图像，其图像文件大小约为 640×480×8≈300KB。

参考答案

（13）B

试题（14）

10000 张分辨率为 1024×768 的真彩（32 位）图片刻录到 DVD 光盘上，假设每张光盘可以存放 4GB 的信息，则需要<u>（14）</u>张光盘。

（14）A. 7　　B. 8　　C. 70　　D. 71

试题（14）分析

本题考查多媒体方面的基础知识。涉及图片存储光盘数量的计算。

图像数据量的计算方式如下：

图像数据量 = 图像的总像素×图像深度　（b）

需用光盘数量的计算方式如下：

光盘数量 = 图像的总像素×图像深度/4GB （张）

参考答案

（14）C

试题（15）

某项目组拟开发一个大规模系统，且具备了相关领域及类似规模系统的开发经验。下列过程模型中，__（15）__最适合开发此项目。

（15）A. 原型模型 B. 瀑布模型 C. V 模型 D. 螺旋模型

试题（15）分析

本题考查软件开发生命周期模型的基本知识。

常见的软件生存周期模型有瀑布模型、演化模型、螺旋模型、喷泉模型等。瀑布模型是将软件生存周期各个活动规定为依线性顺序连接的若干阶段的模型，适合于软件需求很明确的软件项目。V 模型是瀑布模型的一种演变模型，将测试和分析与设计关联进行，加强分析与设计的验证。原型模型是一种演化模型，通过快速构建可运行的原型系统，然后根据运行过程中获取的用户反馈进行改进。演化模型特别适用于对软件需求缺乏准确认识的情况。螺旋模型将瀑布模型和演化模型结合起来，加入了两种模型均忽略的风险分析。

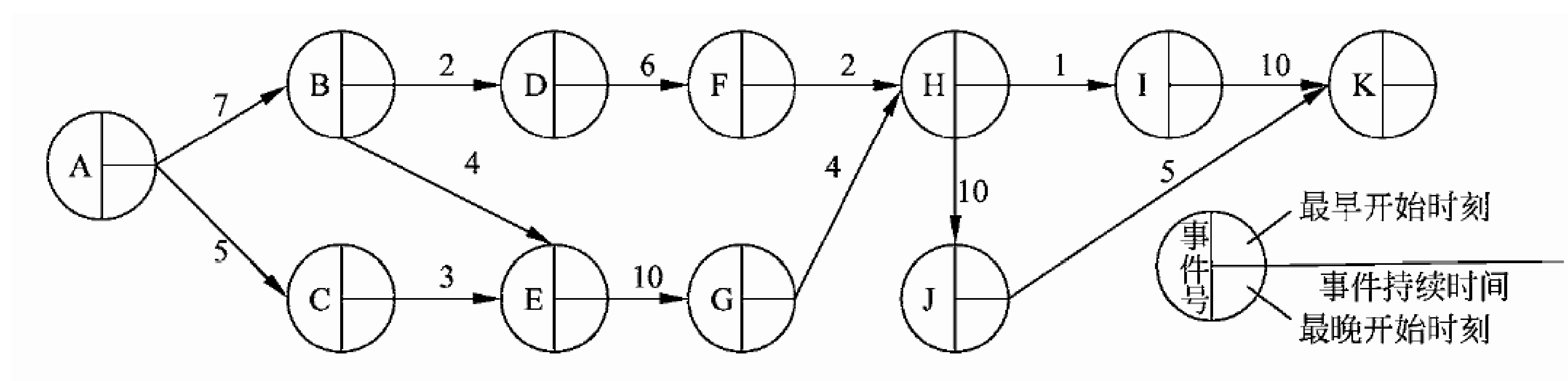
本题中项目组具备了所开发系统的相关领域及类似规模系统的开发经验，即需求明确，瀑布模型最适合开发此项目。

参考答案

（15）B

试题（16）、（17）

使用 PERT 图进行进度安排，不能清晰地描述__（16）__，但可以给出哪些任务完成后才能开始另一些任务。下面的 PERT 图所示工程从 A 到 K 的关键路径是__（17）__（图中省略了任务的开始和结束时刻）。



（16）A. 每个任务从何时开始 B. 每个任务到何时结束
C. 各任务之间的并行情况 D. 各任务之间的依赖关系

（17）A. ABEGHIK B. ABEGHJK C. ACEGHIK D. ACEGHJK

试题（16）、（17）分析

本题考查软件项目管理的基础知识。

软件项目计划的一个重要内容是安排进度，常用的方法有 Gantt 图和 PERT 图。Gantt 图用水平条状图描述，它以日历为基准描述项目任务，可以清楚地表示任务的持续时间和任务之间的并行，但是不能清晰地描述各个任务之间的依赖关系。PERT 图是一种网络模型，描述一个项目的各任务之间的关系。可以明确表达任务之间的依赖关系，即哪些任务完成后才能开始另一些任务，以及如期完成整个工程的关键路径，但是不能清晰地描述各个任务之间的并行关系。

图中任务流 ABEGHIK 的持续时间是 36，ABEGHJK 的持续时间是 40，ACEGHIK 的持续时间是 33，ACEGHJK 的持续时间为 37。所以项目关键路径长度为 40。

参考答案

（16）C　（17）B

试题（18）

敏捷开发方法 XP 是一种轻量级、高效、低风险、柔性、可预测的、科学的软件开发方法，其特性包含在 12 个最佳实践中。系统的设计要能够尽可能早交付，属于＿（18）＿最佳实践。

（18）A. 隐喻　B. 重构　C. 小型发布　D. 持续集成

试题（18）分析

本题考查软件开发过程管理的基本知识。

敏捷开发方法 XP 是一种轻量级、高效、低风险、柔性、可预测的、科学的软件开发方法，其特性包含在 12 个最佳实践中。

（1）计划游戏：快速制定计划、随着细节的不断变化而完善；
（2）小型发布：系统的设计要能够尽可能早地交付；
（3）隐喻：找到合适的比喻传达信息；
（4）简单设计：只处理当前的需求使设计保持简单；
（5）测试先行：先写测试代码再编写程序；
（6）重构：重新审视需求和设计，重新明确地描述它们，以符合新的和现有的需求；
（7）结队编程；
（8）集体代码所有制；
（9）持续集成：可以按日甚至按小时为客户提供可运行的版本；
（10）每周工作 40 个小时；
（11）现场客户；
（12）编码标准。

参考答案

（18）C

试题（19）

在软件开发过程中进行风险分析时，＿（19）＿活动的目的是辅助项目组建立处理风

险的策略，有效的策略应考虑风险避免、风险监控、风险管理及意外事件计划。

（19）A. 风险识别　B. 风险预测　C. 风险评估　D. 风险控制

试题（19）分析

本题考查软件开发风险分析的基本知识。

风险分析实际上是4个不同的活动：风险识别、风险预测、风险评估和风险控制。风险识别是试图系统化地确定对项目计划（估算、进度、资源分配）的威胁。风险预测又称为风险估算，它从两个方面评估一个风险：风险发生的可能性或概率；以及如果风险发生时所产生的后果。风险评估根据风险及其发生的概率和产生的影响预测是否影响参考水平值。风险控制的目的是辅助项目组建立处理风险的策略，有效的策略应考虑风险避免、风险监控、风险管理及意外事件计划。

参考答案

（19）D

试题（20）

以下关于变量和常量的叙述中，错误的是__（20）__。

（20）A. 变量的取值在程序运行过程中可以改变，常量则不行

B. 变量具有类型属性，常量则没有

C. 变量具有对应的存储单元，常量则没有

D. 可以对变量赋值，不能对常量赋值

试题（20）分析

本题考查程序设计语言的基础知识。

变量是计算机内存单元的抽象，在程序中表示数据，具有名称、类型、值、地址、作用域、存储类别等属性，其值在运行过程中由指令进行修改。常量也用于在程序中表示数据，但常量在程序运行过程中不能修改，常量也具有类型，如整型常量、浮点型常量、字符串常量等，也称为字面量或文字。

参考答案

（20）B

试题（21）

编译程序分析源程序的阶段依次是__（21）__。

（21）A. 词法分析、语法分析、语义分析　B. 语法分析、词法分析、语义分析

C. 语义分析、语法分析、词法分析　D. 语义分析、词法分析、语法分析

试题（21）分析

本题考查程序语言翻译的基础知识。

编译程序是一种将高级语言程序翻译成目标程序的系统软件，它对源程序的翻译过程分为词法分析、语法分析、语义分析、中间代码生成、代码优化和目标代码生成，以及符号表管理和出错处理。

源程序可以被看成是一个字符串。词法分析是编译过程的第一阶段，其任务是对源

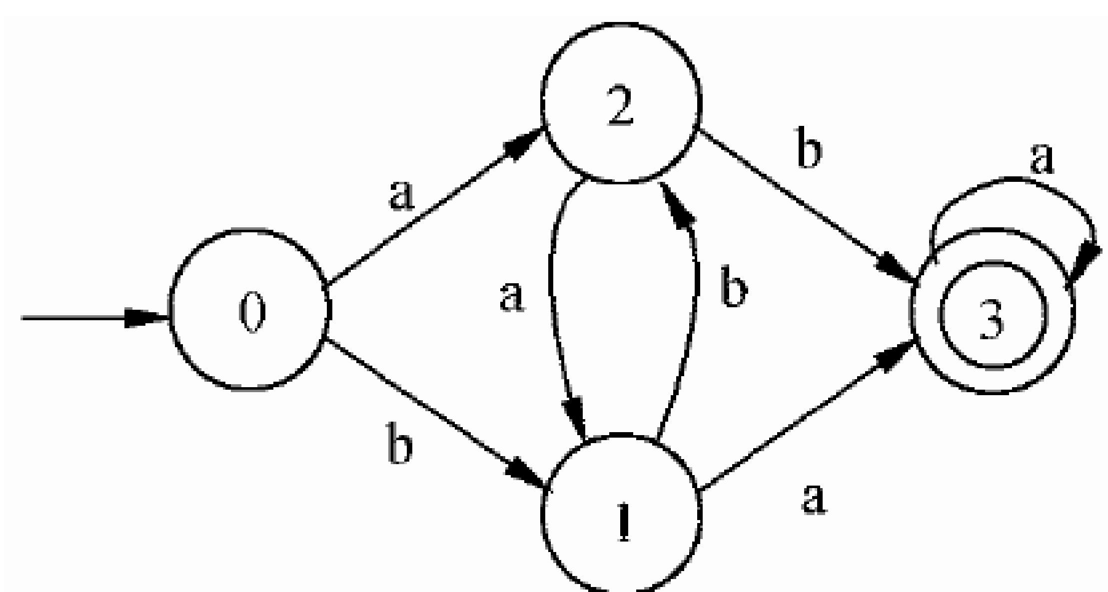
程序从前到后（从左到右）逐个字符地扫描，从中识别出一个个的“单词”符号。语法分析的任务是在词法分析的基础上，根据语言的语法规则将单词符号序列分解成各类语法单位，如“表达式”“语句”“程序”等。语义分析阶段主要检查源程序是否包含语义错误，并收集类型信息供后面的代码生成阶段使用。只有语法和语义都正确的源程序才能被翻译成正确的目标代码。

参考答案

（21）A

试题（22）

下图所示的有限自动机中，0 是初始状态，3 是终止状态，该自动机可以识别<u>（22）</u>。



（22）A. abab　　B. aaaa　　C. bbbb　　D. abba

试题（22）分析

本题考查程序语言翻译的基础知识。

有限自动机可识别一个字符串的含义是，从有限自动机的初态出发，存在一条到达终态的路径，其上的标记可构成该字符串。若从初态到终态不存在能构成指定字符串的路径，则称该字符串不能被该自动机识别。

对于“abab”，其识别路径为状态 0→状态 2→状态 3→状态 3，虽然到达终态，但是没有识别出最后的字符“b”。

对于“bbbb”，其识别路径为状态 0→状态 1→状态 2→状态 3，虽然到达终态，但是没有识别出最后的字符“b”。

对于“abba”，其识别路径为状态 0→状态 2→状态 3，虽然到达终态，但是没有识别出“ba”。

对于“aaaa”，其识别路径为状态 0→状态 2→状态 1→状态 3→状态 3，存在从初态到终态的路径标记形成“aaaa”，所以可识别。

参考答案

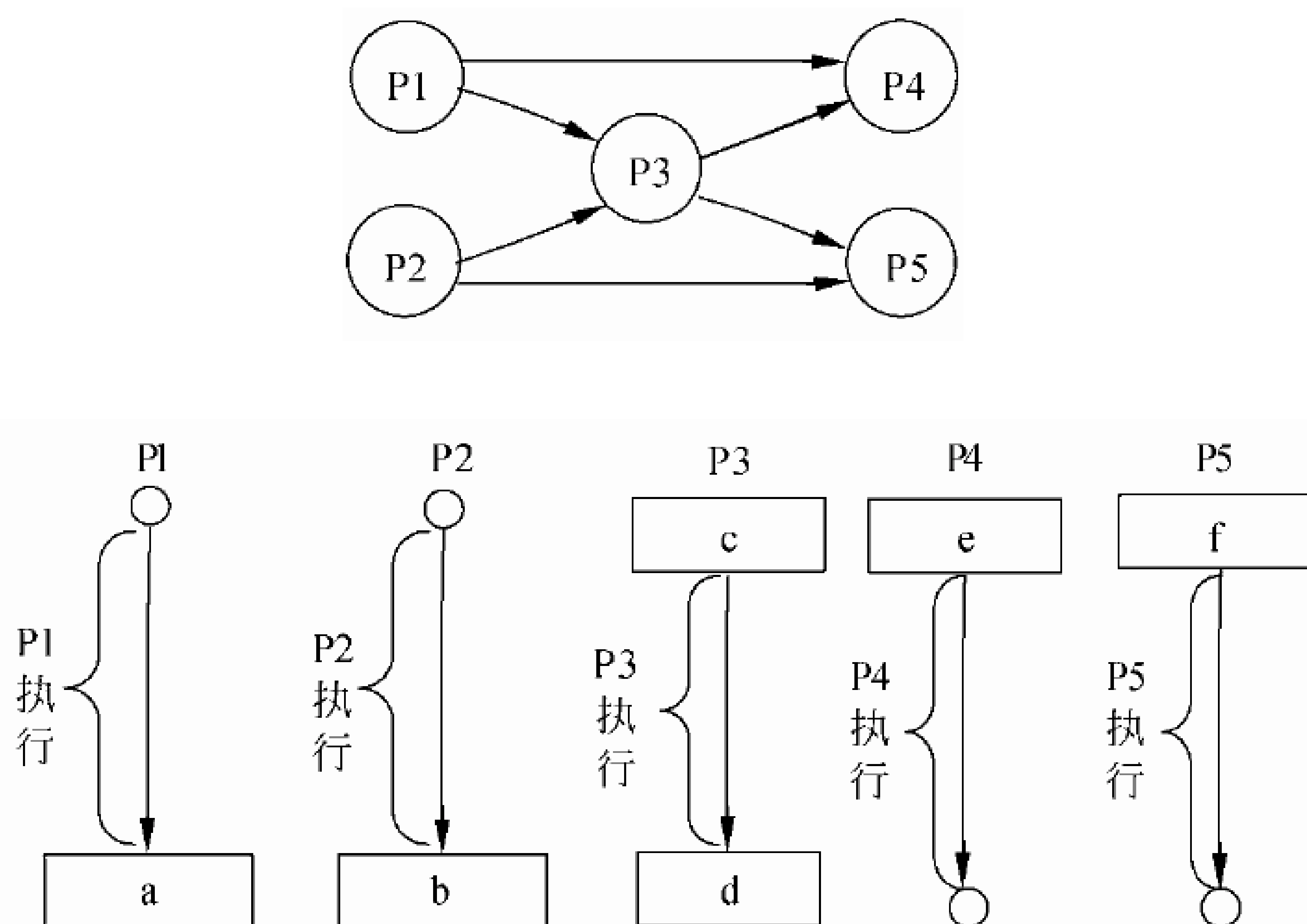
（22）B

试题（23）～（25）

进程 P1、P2、P3、P4 和 P5 的前趋图如下：

若用 PV 操作控制进程 P1～P5 并发执行的过程，则需要设置 6 个信号量 S1、S2、

S3、S4、S5 和 S6，且信号量 S1～S6 的初值都等于零。下图中 a 和 b 处应分别填写__（23）__；c 和 d 处应分别填写__（24）__，e 和 f 处应分别填写__（25）__。



（23）A. P(S1) P(S2) 和 P(S3) P(S4)　　B. P(S1) V(S2) 和 P(S2) V(S1)
　　C. V(S1) V(S2) 和 V(S3) V(S4)　　D. P(S1) P(S2) 和 V(S1) V(S2)

（24）A. P(S1) P(S2) 和 V(S3) V(S4)　　B. P(S1) P(S3) 和 V(S5) V(S6)
　　C. V(S1) V(S2) 和 P(S3) P(S4)　　D. P(S1) V(S3) 和 P(S2) V(S4)

（25）A. P(S3) P(S4) 和 V(S5) V(S6)　　B. V(S5) V(S6) 和 P(S5) P(S6)
　　C. P(S2) P(S5) 和 P(S4) P(S6)　　D. P(S4) V(S5) 和 P(S5) V(S6)

试题（23）～（25）分析

本题考查 PV 操作方面的基本知识。

试题（23）的正确答案是 C，因为 P1 是 P3 和 P4 的前驱，当 P1 执行完成后，应通知 P3 和 P4，故应采用 V(S1) V(S2)操作分别通知 P3 和 P4；同理，P2 是 P3 和 P5 的前驱，当 P2 执行完后，应通知 P3 和 P5，故应采用 V(S3) V(S4)操作分别通知 P3 和 P5。

试题（24）的正确答案是 B，因为 P3 是 P1 和 P2 的后继，当 P3 执行前应测试 P1 和 P2 是否执行完，故应采用 P(S1) P(S3)操作分别测试 P1 和 P2 是否执行完；又因为 P3 是 P4 和 P5 的前驱，当 P3 执行完应通知 P4 和 P5，故应采用 V(S5) V(S6)操作分别通知 P4 和 P5。

试题（25）的正确答案是 C，因为 P4 是 P1 和 P3 的后继，当 P4 执行前应测试 P1 和 P3 是否执行完，故应采用 P(S2) P(S5)操作分别测试 P1 和 P3 是否执行完；又因为 P5 是 P2 和 P3 的前驱的后继，当 P5 执行前应测试 P2 和 P3 是否执行完，故应采用 P(S4) P(S6)操作分别测试 P2 和 P3 是否执行完。

参考答案

（23）C　（24）B　（25）C

试题（26）

某磁盘磁头从一个磁道移至另一个磁道需要 10ms。文件在磁盘上非连续存放，逻辑上相邻数据块的平均移动距离为 10 个磁道，每块的旋转延迟时间及传输时间分别为 100ms 和 2ms，则读取一个 100 块的文件需要 __（26）__ ms 时间。

（26）A. 10200　B. 11000　C. 11200　D. 20200

试题（26）分析

本题考查操作系统中设备管理的基本知识。

访问一个数据块的时间应为寻道时间加旋转延迟时间及传输时间。根据题意，每块的旋转延迟时间及传输时间共需 102ms，磁头从一个磁道移至另一个磁道需要 10ms，但逻辑上相邻数据块的平均距离为 10 个磁道，即读完一个数据块到下一个数据块寻道时间需要 100ms。通过上述分析，本题访问一个数据块的时间应为 202ms，而读取一个 100 块的文件共需要 20200ms，因此，本题的正确答案为 D。

参考答案

（26）D

试题（27）、（28）

某文件系统采用多级索引结构，若磁盘块的大小为 512B，每个块号需占 3B，那么根索引采用一级索引时的文件最大长度为 __（27）__ KB；采用二级索引时的文件最大长度为 __（28）__ KB。

（27）A. 85　B. 170　C. 512　D. 1024

（28）A. 512　B. 1024　C. 14450　D. 28900

试题（27）、（28）分析

本题考查操作系统中文件管理的基本知识。

根据题意，磁盘块的大小为 512B，每个块号需占 3B，因此一个磁盘物理块可存放 512/3=170 个块号。

根索引采用一级索引时的文件最大长度为：

$$170\times512/1024=87040/1024=85KB$$

根索引采用二级索引时的文件最大长度为：

$$170\times170\times512/1024=28900\times512/1024=14450KB$$

参考答案

（27）A　（28）C

试题（29）

冗余技术通常分为 4 类，其中 __（29）__ 按照工作方法可以分为静态、动态和混合

冗余。

（29）A. 时间冗余　　B. 信息冗余　　C. 结构冗余　　D. 冗余附加技术

试题（29）分析

冗余是指对于实现系统规定功能是多余的那部分资源，包括硬件、软件、信息和时间。通常冗余技术分为 4 类：（1）结构冗余，按其工作方法可以分为静态、动态和混合冗余；（2）信息冗余，指的是为了检测或纠正信息在运算或传输中的错误另外加的一部分信息；（3）时间冗余，是指以重复执行指令或程序来消除瞬时错误带来的影响；（4）冗余附件技术，是指为实现上述冗余技术所需的资源和技术。

参考答案

（29）C

试题（30）

以下关于过程改进的叙述中，错误的是__（30）__。

（30）A. 过程能力成熟度模型基于这样的理念：改进过程将改进产品，尤其是软件产品

B. 软件过程改进框架包括评估、计划、改进和监控 4 个部分

C. 软件过程改进不是一次性的，需要反复进行

D. 在评估后要把发现的问题转化为软件过程改进计划

试题（30）分析

软件成熟度模型 CMM 是对软件组织进化阶段的描述，该模型在解决软件过程存在问题方面取得了很大的成功，因此在软件界产生了巨大影响，促使软件界重视并认真对待过程改进工作。过程能力成熟度模型基于这样的理念：改进过程将改进产品，尤其是软件产品。软件组织为提高自身的过程能力，把不够成熟的过程提升到较成熟的过程涉及 4 个方面，这 4 个方面构成了软件过程改进的框架，即过程改进基础设施、过程改进线路图、软件过程评估方法和软件过程改进计划。在进行评估后需要把发现的问题转化为软件过程改进计划。而过程改进通常不可能是一次性的，需要反复进行。每一次改进要经历 4 个步骤：评估、计划、改进和监控。

参考答案

（30）B

试题（31）

软件复杂性度量的参数不包括__（31）__。

（31）A. 软件的规模　　B. 开发小组的规模

C. 软件的难度　　D. 软件的结构

试题（31）分析

软件复杂性度量是软件度量的一个重要分支。软件复杂性度量的参数有很多，主要包括：（1）规模，即指令数或者源程序行数；（2）难度，通常由程序中出现的操作数所

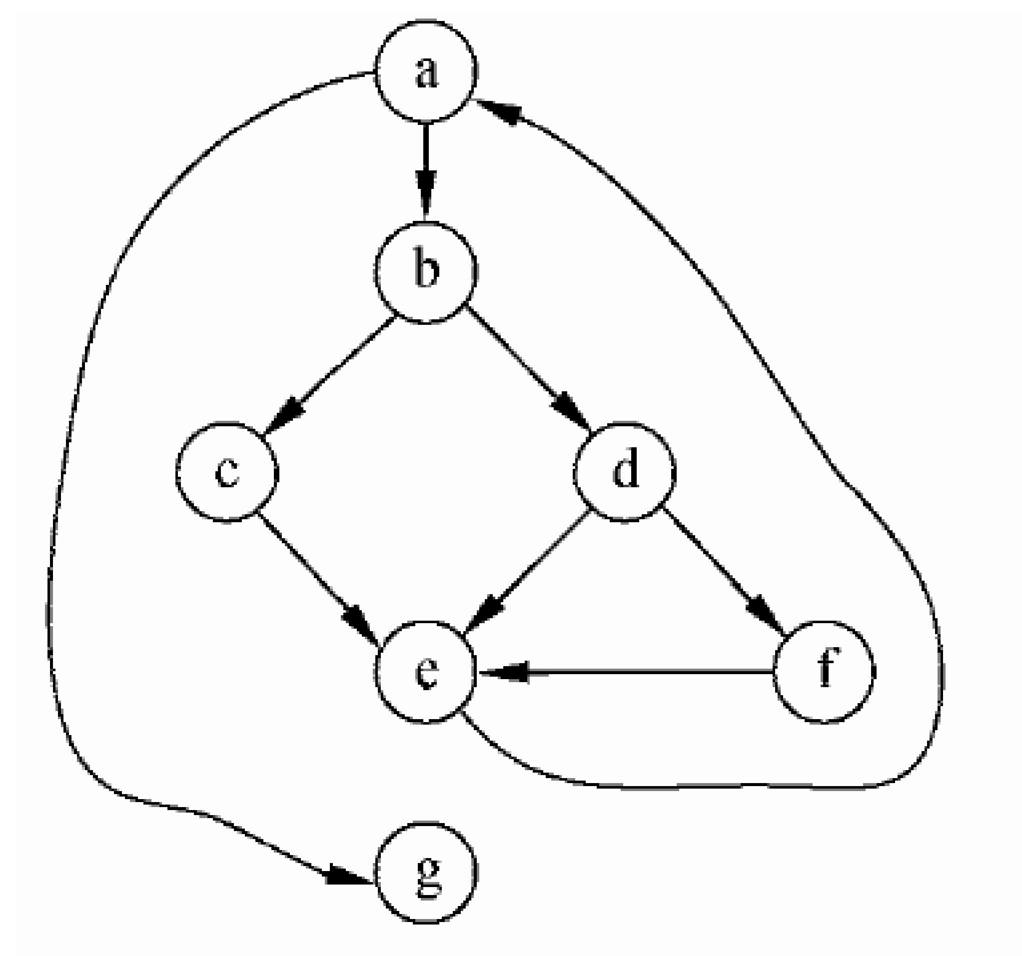
决定的量来表示；（3）结构，通常用与程序结构有关的度量来表示；（4）智能度，即算法的难易程度。

参考答案

（31）B

试题（32）

根据 McCabe 度量法，以下程序图的复杂性度量值为__(32)__。



（32）A. 4　　B. 5　　C. 6　　D. 7

试题（32）分析

软件复杂性度量是软件度量的一个重要分支，而其主要表现在程序的复杂性。其中，McCabe 度量法是一种基于程序控制流的复杂性度量方法，该方法认为程序的复杂性很大程度上取决于控制的复杂性。首先根据程序画出程序图，然后基于图论用图的环路数来度量程序复杂性，即 $V(G)=m-n+2p$，其中 m，n 和 p 分别表示图 G 中弧的个数、顶点的个数和强连通分量数。根据上述公式可得，上图的复杂性为 9–7+2=4。

参考答案

（32）A

试题（33）

软件系统的可维护性评价指标不包括__(33)__。

（33）A. 可理解性　　B. 可测试性　　C. 可扩展性　　D. 可修改性

试题（33）分析

软件的可维护性是指维护人员理解、改正、改动和改进这个软件的难易程度，是软件开发阶段各个时期的关键目标。软件系统的可维护性评价指标包括可理解性、可测试性、可修改性、可靠性、可移植性、可使用性和效率。

参考答案

（33）C

试题（34）

以下关于软件系统文档的叙述中，错误的是__（34）__。

（34）A. 软件系统文档既包括有一定格式要求的规范文档，又包括系统建设过程中的各种来往文件、会议纪要、会计单据等资料形成的不规范文档

B. 软件系统文档可以提高软件开发的可见度

C. 软件系统文档不能提高软件开发效率

D. 软件系统文档便于用户理解软件的功能、性能等各项指标

试题（34）分析

软件系统文档是系统建设过程的“痕迹”，是系统维护人员的指南，是开发人员与用户交流的工具。软件系统文档不仅包括应用软件开发过程中产生的文档，还包括硬件采购和网络设计中形成的文档；不仅包括有一定格式要求的规范文档，还包括系统建设过程中的各种来往文件、会议纪要、会计单据等资料形成的不规范文档。软件系统文档可以提高软件开发的可见度，提高软件开发效率，且便于用户理解软件的功能、性能等各项指标。

参考答案

（34）C

试题（35）

以下关于软件测试的叙述中，正确的是__（35）__。

（35）A. 软件测试不仅能表明软件中存在错误，也能说明软件中不存在错误

B. 软件测试活动应从编码阶段开始

C. 一个成功的测试能发现至今未发现的错误

D. 在一个被测程序段中，若已发现的错误越多，则残存的错误数越少

试题（35）分析

软件测试是软件开发过程中一个独立而且非常重要的阶段，它是为了发现错误而执行程序的过程。因此一个成功的测试应该能发现至今未发现的错误。而且需要特别指出的是软件测试不能表明软件中不存在错误，它只能说明软件中存在错误。另外，由于问题的复杂性、软件本身的复杂性和抽象性、软件开发各个阶段工作的多样性、参加开发各种人员之间的配合关系等因素，使得开发的每个环节都可能产生错误，因此软件测试应该贯穿到软件开发的各个阶段中，且需要尽早地和不断地进行。经验表明，测试中存在一种集群现象，即在被测程序段中，若发现的错误数目越多，则残存的错误数目也较多。

参考答案

（35）C

试题（36）

不属于黑盒测试技术的是__（36）__。

（36）A. 错误猜测　B. 逻辑覆盖　C. 边界值分析　D. 等价类划分

试题（36）分析

黑盒测试也称为功能测试，在完全不考虑软件的内部结构和特性的情况下来测试软件的外部特性。常用的黑盒测试技术包括等价类划分、边界值分析、错误猜测和因果图的报告。白盒测试也称为结构测试，根据程序的内部结构和逻辑来设计测试用例，对程序的执行路径和过程进行测试，检查是否满足设计的需要。常用的白盒测试技术包括逻辑覆盖和基本路径测试。

参考答案

（36）B

试题（37）～（42）

开-闭原则（Open-Closed Principle, OCP）是面向对象的可复用设计的基石。开-闭原则是指一个软件实体应当对__（37）__开放，对__（38）__关闭；里氏代换原则（Liskov Substitution Principle, LSP）是指任何__（39）__可以出现的地方，__（40）__一定可以出现。依赖倒转原则（Dependence Inversion Principle, DIP）就是要依赖于__（41）__，而不依赖于__（42）__，或者说要针对接口编程，不要针对实现编程。

（37）A. 修改　B. 扩展　C. 分析　D. 设计
（38）A. 修改　B. 扩展　C. 分析　D. 设计
（39）A. 变量　B. 常量　C. 基类对象　D. 子类对象
（40）A. 变量　B. 常量　C. 基类对象　D. 子类对象
（41）A. 程序设计语言　B. 建模语言　C. 实现　D. 抽象
（42）A. 程序设计语言　B. 建模语言　C. 实现　D. 抽象

试题（37）～（42）分析

本题考查面向对象设计的原则。

开-闭原则（Open-Closed Principle）是面向对象的可复用设计（Object Oriented Design，OOD）的基石。开-闭原则是指一个软件实体应当对扩展开放，对修改关闭，即在设计一个模块的时候，应当使这个模块可以在不被修改的前提下被扩展。满足开-闭原则的系统可以通过扩展已有的软件系统，提供新的能力和行为，以满足对软件的新需求，使软件系统有一定的适应性和灵活性；因为已有的软件模块，特别是最重要的抽象层模块不能再修改，这就使变化中的软件系统有一定的稳定性和延续性；满足开-闭原则的系统具备更好的可复用性与可维护性。

在面向对象编程中，通过抽象类及接口，规定了具体类的特征作为抽象层，相对稳定，从而满足“对修改关闭”的要求；而从抽象类导出的具体类可以改变系统的行为，从而满足对扩展开放。

里氏代换原则（Liskov Substitution Principle, LSP）是指一个软件实体如果使用的是一个基类的话，那么一定适用于其子类，而且软件系统觉察不出基类对象和子类对象的

区别，也就是说，在软件系统中把基类都替换成它的子类，程序的行为没有变化。但需要注意的是，里氏代换原则中仅仅指出了用子类的对象去代替基类的对象，而反过来的代换则是不成立的。例如，如果一个软件模块中使用的是一个子类对象，那么使用父类对象去代换子类对象则可能产生错误。用一句简单的话概括：任何基类对象可以出现的地方，子类对象一定可以代替基类对象。

依赖倒转原则（Dependence Inversion Principle，DIP）就是要依赖于抽象，而不依赖于实现，或者说要针对接口编程，不要针对实现编程。系统中进行设计和实现的时候应当使用接口和抽象类进行变量类型声明、参数类型声明、方法返回类型说明，以及数据类型的转换等，而不要用具体类进行上述操作。要保证做到这一点，一个具体类应当只实现接口和抽象类中声明过的方法，而不要给出多余的方法。

传统的过程性系统的设计办法倾向于使高层次的模块依赖于低层次的模块，抽象层次依赖于具体层次。依赖倒转原则就是把这个不良的依赖关系倒转过来。面向对象设计的重要原则是创建抽象层次，并且从该抽象层次导出具体层次，具体层次给出不同的实现。继承关系就是一种从抽象化到具体化的导出。抽象层包含的应该是应用系统的业务逻辑和宏观的、对整个系统来说重要的战略性决定，而具体层次含有的是一些次要的与实现有关的算法和逻辑，以及战术性的决定，带有一定的偶然性选择。从复用的角度来说，高层抽象的模块是应当复用的，而且是复用的重点，因为它含有一个应用系统最重要的宏观业务逻辑，是较为稳定的部分。而在传统的过程性设计中，复用则侧重于具体层次模块的复用。

使用依赖倒转原则时建议不依赖于具体类，即程序中所有的依赖关系都应该终止于抽象类或者接口。尽量做到：任何变量都不应该持有一个指向具体类的指针或者引用；任何类都不应该从具体类派生；任何方法都不应该覆写它的任何基类中的已经实现的方法。

参考答案

（37）B　　（38）A　　（39）C　　（40）D　　（41）D　　（42）C

试题（43）～（45）

__（43）__是一种很强的“拥有”关系，“部分”和“整体”的生命周期通常一样。整体对象完全支配其组成部分，包括它们的创建和销毁等；__（44）__同样表示“拥有”关系，但有时候“部分”对象可以在不同的“整体”对象之间共享，并且“部分”对象的生命周期也可以与“整体”对象不同，甚至“部分”对象可以脱离“整体”对象而单独存在。上述两种关系都是__（45）__关系的特殊种类。

（43）A. 聚合　　B. 组合　　C. 继承　　D. 关联
（44）A. 聚合　　B. 组合　　C. 继承　　D. 关联
（45）A. 聚合　　B. 组合　　C. 继承　　D. 关联

试题（43）～（45）分析

本题考查组合和聚合的基本概念。

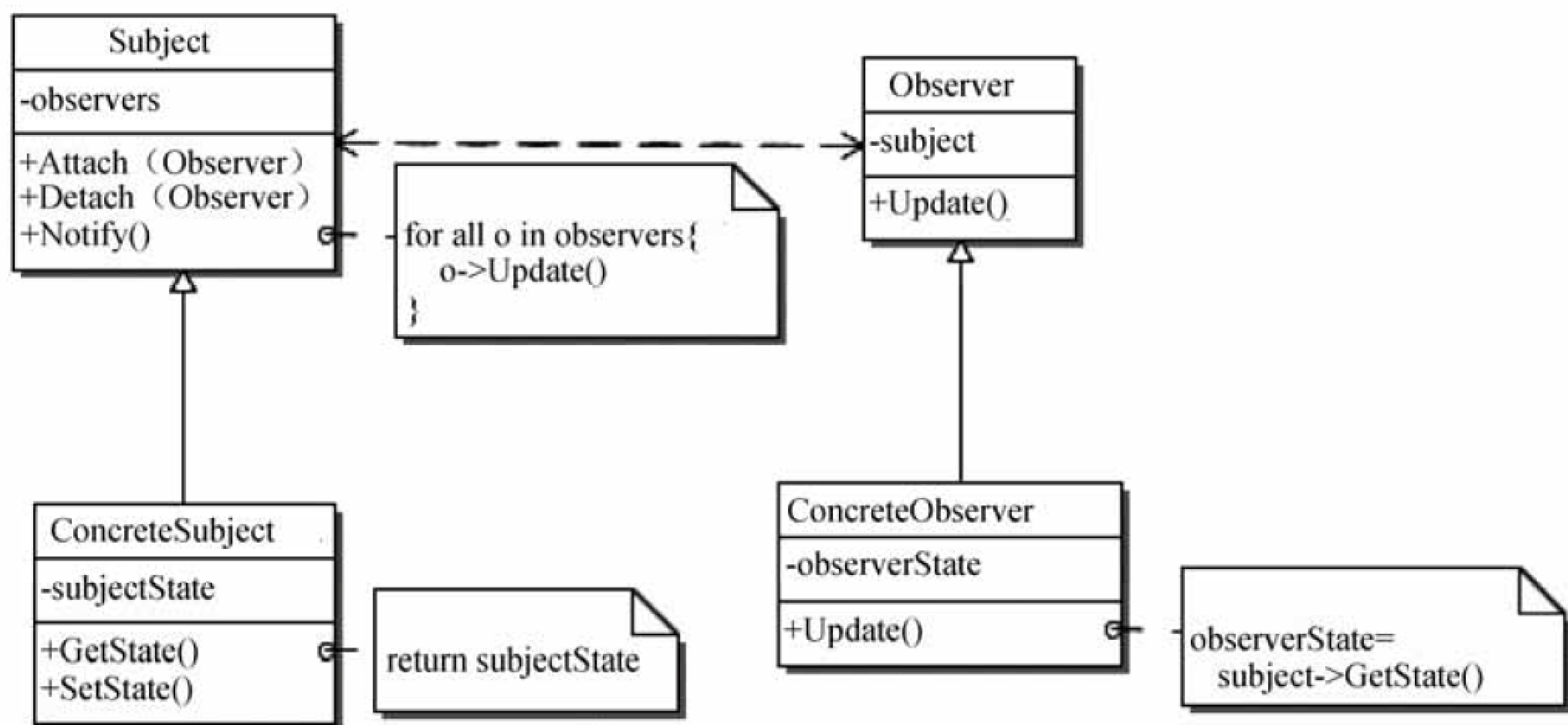
组合（Composition）和聚合（Aggregation）都是关联（Association）的特殊种类。组合是一种很强的“拥有”关系，部分和整体的生命周期通常一样。组合成的新对象完全支配其组成部分，包括它们的创建和湮灭等。一个组合关系的成分对象是不能被另一个组合构成的对象共享的。聚合同样表示“拥有”关系，但其程度不如组合强，有时候“部分”对象可以在不同的“整体”对象之间共享，并且“部分”对象的生命周期也可以与“整体”对象不同，甚至“部分”对象可以脱离“整体”对象而单独存在。一般而言，组合是值的合成（Aggregation by Value），而聚合是引用的合成（Aggregation by Reference）。

参考答案

（43）B　　（44）A　　（45）D

试题（46）、（47）

下面的 UML 类图描绘的是__（46）__设计模式。关于该设计模式的叙述中，错误的是__（47）__。



（46）A. 桥接　　B. 策略　　C. 抽象工厂　　D. 观察者

（47）A. 该设计模式中的 Observer 需要维护至少一个 Subject 对象

B. 该设计模式中的 ConcreteObserver 可以绕过 Subject 及其子类的封装

C. 该设计模式中一个 Subject 对象需要维护多个 Observer 对象

D. 该设计模式中 Subject 需要通知 Observer 对象其自身的状态变化

试题（46）、（47）分析

本题考查面向对象设计中的设计模式。

题中的类图是观察者设计模式，在该设计模式中的 Subject 和 Observer 分别表示抽

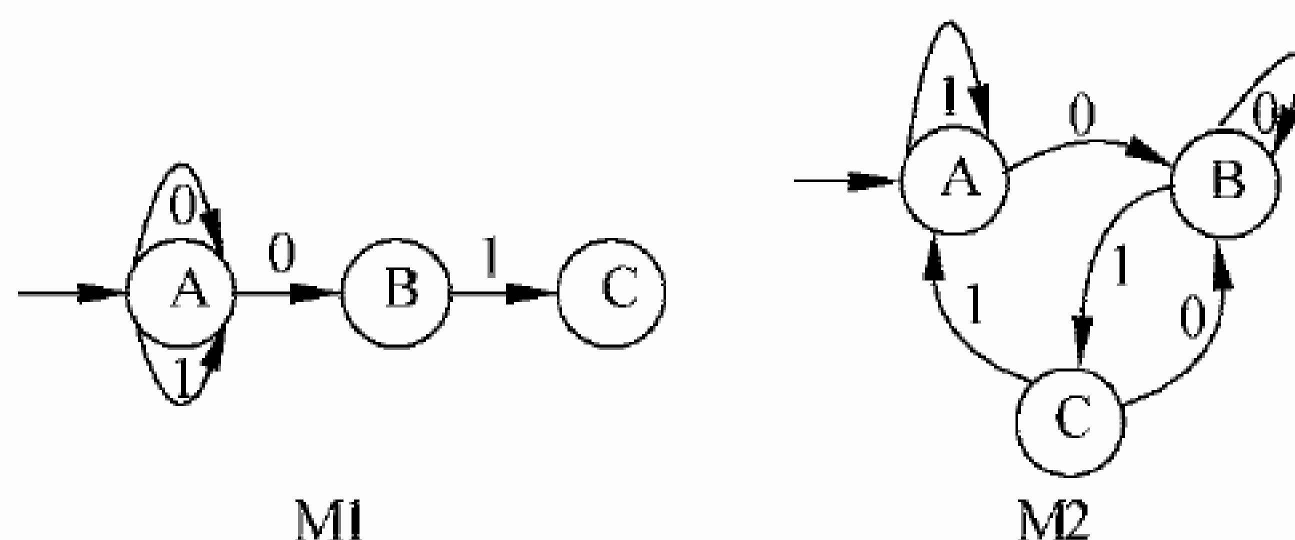
象的被观察者和观察者。通常，一个观察者（Observer）观察一个被观察者（Subject），而一个被观察者可以被多个观察者关注。当 Subject 的状态发生变化时，Subject 将通知所有的 Observer，告知状态已经发生了变化，而 Observer 收到通知后，将查询 Subject 的状态。

参考答案

（46）D （47）B

试题（48）

下图所示为两个有限自动机 M1 和 M2（A 是初态、C 是终态），__（48）__。



（48）A. M1 和 M2 都是确定的有限自动机

B. M1 和 M2 都是不确定的有限自动机

C. M1 是确定的有限自动机，M2 是不确定的有限自动机

D. M1 是不确定的有限自动机，M2 是确定的有限自动机

试题（48）分析

本题考查程序语言翻译的基础知识。

有限自动机是一种识别装置的抽象概念，它能准确地识别正规集。有限自动机分为两类：确定的有限自动机和不确定的有限自动机。它们都可以用状态转换图和状态转换矩阵表示。

一个确定的有限自动机是个五元组：（S，Σ，f，$s0$，Z），其中：S 是一个有限集合，它的每个元素称为一个状态；Σ 是一个有穷字母表，它的每个元素称为一个输入字符；f 是从 $S\times\Sigma\rightarrow S$ 上的单值部分映像，f（A，a）$=Q$ 表示当前状态为 A、输入为 a 时，将转换到下一状态 Q。我们称 Q 为 A 的一个后继状态；$s0\in S$，是唯一的一个开始状态；Z 是非空的终止状态集合，$Z\subseteq S$。

一个不确定的有限自动机也是一个五元组，它与确定有限自动机的区别是：f 是从 $S\times\Sigma\rightarrow 2S$ 上的映像。对于 S 中的一个给定状态及输入符号，返回一个状态的集合。即当前状态的后继状态不一定是唯一确定的；有向弧上的标记可以是 ε。

参考答案

（48）D

试题（49）

以下关于可视化程序设计的叙述中，错误的是__（49）__。

（49）A. 可视化程序设计使开发应用程序无须编写程序代码

B. 可视化程序设计基于面向对象的思想，引入了控件和事件驱动

C. 在可视化程序设计中，构造应用程序界面就像搭积木

D. 在可视化程序设计中，采用解释方式可随时查看程序的运行效果

试题（49）分析

本题考查程序设计的基础知识。

可视化程序设计是以“所见即所得”的编程思想为原则，力图实现编程工作的可视化，即随时可以看到结果，程序与结果的调整同步。

与传统的编程方式相比，“可视化程序设计”仅通过直观的操作方式即可完成界面的设计工作。

可视化程序设计语言的特点主要表现在两个方面：一是基于面向对象的思想，引入了控件的概念和事件驱动；二是程序开发过程一般遵循以下步骤，即先进行界面的绘制工作，再基于事件编写程序代码，以响应鼠标、键盘的各种动作。

可视化程序设计最大的优点是设计人员可以不用编写或只需编写很少的程序代码，就能完成应用程序的设计，这样就能极大地提高设计人员的工作效率。

参考答案

（49）A

试题（50）

以下关于汇编语言的叙述中，错误的是__（50）__。

（50）A. 汇编语言源程序中的指令语句将被翻译成机器代码

B. 汇编程序先将源程序中的伪指令翻译成机器代码，然后再翻译指令语句

C. 汇编程序以汇编语言源程序为输入，以机器语言表示的目标程序为输出

D. 汇编语言的指令语句必须具有操作码字段，可以没有操作数字段

试题（50）分析

本题考查程序设计语言的基础知识。

汇编语言源程序中的每一条指令语句在源程序汇编时都要产生可供计算机执行的指令代码（即目标代码）。

伪指令语句用于指示汇编程序如何汇编源程序，常用于为汇编程序提供以下信息：该源程序如何分段，有哪些逻辑段在程序段中，哪些是当前段，它们分别由哪个段寄存器指向；定义了哪些数据，存储单元是如何分配的等。伪指令语句除定义的具体数据要生成目标代码外，其他均没有对应的目标代码。伪指令语句的这些命令功能是由汇编程序在汇编源程序时，通过执行一段程序来完成的，而不是在运行目标程序时实现的。

目前主要有两种不同标准的汇编语言指令格式：Windows 下的汇编语言基本上都遵循 Intel 风格的语法，如 MASM、NASM，而 Unix/Linux 下的汇编语言基本上都遵循 AT&T 风格的语法。

汇编语言语句的通用格式如下：

```
[名称[:]] 指令码 [第一操作数][,第二操作数] ;注释
```

汇编语言指令码的操作数的个数可以是 0、1、2 个；当操作数的个数为 2 的时候，语句还有两种不同的格式。

```
Windows 下 Intel 风格的汇编语言语句格式为：
    [名称[:]] 指令码 目的操作数 DST, 源操作数 SRC ;注释
Unix/Linux 下 AT&T 风格的汇编语言语句格式为：
    [名称[:]] 指令码 源操作数 SRC, 目的操作数 DST ;注释
```

汇编语言语句格式中的“名称”并不是所有语句都必需的。如果语句中带有“名称”，则大多数情况下“名称”都表示的是内存中某一存储单元的地址，也就是其后面各项在内存中存放的第一个存储单元的地址。

参考答案

（50）B

试题（51）

在某企业的营销管理系统设计阶段，属性“员工”在考勤管理子系统中被称为“员工”，而在档案管理子系统中被称为“职工”，这类冲突称为__（51）__冲突。

（51）A. 语义　　B. 结构　　C. 属性　　D. 命名

试题（51）分析

本题考查数据库概念结构设计中的基础知识。根据局部应用设计好各局部 E-R 图之后，就可以对各个局部 E-R 图进行合并。合并的目的在于解决分 E-R 图中相互间存在的冲突，消除分 E-R 图之间存在的信息冗余，使之成为能够被全系统所有用户共同理解和接受的统一的、精炼的全局概念模型。分 E-R 图之间的冲突主要分为结构冲突、属性冲突和命名冲突三类。

选项 A 显然是不正确的。

选项 B 不正确。因为结构冲突是指同一实体在不同的分 E-R 图中有不同的属性，同一对象在某一分 E-R 图中被抽象为实体而在另一分 E-R 图中又被抽象为属性，需要统一。

选项 C 不正确，因为属性冲突是指同一属性可能会存在于不同的分 E-R 图，由于设计人员不同或是出发点不同，对属性的类型、取值范围、数据单位等可能会不一致，这些属性对应的数据将来只能以一种形式在计算机中存储，这就需要在设计阶段进行统一。

选项 D 正确，因为命名冲突是指相同意义的属性在不同的分 E-R 图上有着不同的命名，或是名称相同的属性在不同的分 E-R 图中代表着不同的意义，这些也要进行统一。

参考答案

（51）D

试题（52）、（53）

设有学生实体 Students（学号，姓名，性别，年龄，家庭住址，家庭成员，关系，联系电话），其中“家庭住址”记录了邮编、省、市、街道信息；“家庭成员，关系，联系电话”分别记录了学生亲属的姓名、与学生的关系以及联系电话。

学生实体 Students 中的“家庭住址”是一个<u>（52）</u>属性；为使数据库模式设计更合理，对于关系模式 Students<u>（53）</u>。

（52）A. 简单　　B. 多值　　C. 复合　　D. 派生

（53）A. 可以不作任何处理，因为该关系模式达到了 3NF

B. 只允许记录一个亲属的姓名、与学生的关系以及联系电话的信息

C. 需要对关系模式 Students 增加若干组家庭成员、关系及联系电话字段

D. 应该将家庭成员、关系及联系电话加上学生号，设计成为一个独立的实体

试题（52）、（53）分析

本题考查关系运算和 E-R 图的基本概念。

试题（52）的正确答案为 C。简单属性是原子的、不可再分的。复合属性可以细分为更小的部分（即划分为别的属性）。有时用户希望访问整个属性，有时希望访问属性的某个成分，那么在模式设计时可采用复合属性。本题学生实体集 Students 的“家庭住址”可以进一步分为邮编、省、市、街道。

在大多数情况下，定义的属性对于一个特定的实体都只有单独的一个值。例如，对于一个特定的学生，只对应一个学生号、学生姓名，这样的属性叫做单值属性。但是，在某些特定情况下，一个属性可能对应一组值。例如，学生可能有 0 个、1 个或多个亲属，那么学生的亲属的姓名可能有多个。这样的属性称为多值属性。为了将数据库模式设计得更合理，试题（53）应该将家庭成员、关系及联系电话加上学生号设计成为一个独立的实体。

参考答案

（52）C　（53）D

试题（54）～（56）

设有关系模式 *R*（课程，教师，学生，成绩，时间，教室），其中函数依赖集 *F* 如下：

F={课程→→教师，（学生，课程）→成绩，（时间，教室）→课程，
（时间，教师）→教室，（时间，学生）→教室}

关系模式 *R* 的一个主键是<u>（54）</u>，R 规范化程度最高达到<u>（55）</u>。若将关系模式 *R* 分解为 3 个关系模式 *R*1（课程，教师）、*R*2（学生，课程，成绩）、*R*3（学生，时间，教室，课程），其中 *R*2 的规范化程度最高达到<u>（56）</u>。

（54）A.（学生，课程）　　B.（时间，教室）

C.（时间，教师）　　D.（时间，学生）

（55）A. 1NF　　B. 2NF　　C. 3NF　　D. BCNF

（56）A. 2NF　　B. 3NF　　C. BCNF　　D. 4NF

试题（54）～（56）分析

本题主要考查关系模式规范化方面的相关知识。

试题（54）的正确答案为 D。因为根据函数依赖集 *F* 可知（时间，学生）可以决定关系 R 中的全部属性，故关系模式 *R* 的一个主键是（时间，学生）。

试题（55）的正确答案为 B。因为根据函数依赖集 *F* 可知，*R* 中的每个非主属性完全函数依赖于（时间，学生），所以 *R* 是 2NF。

试题（56）的正确答案为 C。因为 *R*2（学生，课程，成绩）的主键为（学生，课程），而 *R*2 的每个属性都不传递依赖于 *R*2 的任何键，所以 *R*2 是 BCNF。

参考答案

（54）D　　（55）B　　（56）C

试题（57）

设循环队列 Q 的定义中有 rear 和 len 两个域变量，其中 rear 表示队尾元素的指针，len 表示队列的长度，如下图所示（队列长度为 3，队头元素为 e）。设队列的存储空间容量为 M，则队头元素的指针为（57）。

（57）A. (Q.rear+Q.len−1)　　B. (Q.rear+Q.len−1+M)%M

C. (Q.rear−Q.len+1)　　D. (Q.rear−Q.len+1+M)%M

试题（57）分析

本题考查数据结构的基础知识。

从题目中的图可以推导出，队头元素的指针为(Q.rear−Q.len+1+M)%M。

参考答案

（57）D

试题（58）

下面关于哈夫曼树的叙述中，正确的是（58）。

（58）A. 哈夫曼树一定是完全二叉树

B. 哈夫曼树一定是平衡二叉树

C. 哈夫曼树中权值最小的两个节点互为兄弟节点

D. 哈夫曼树中左孩子节点小于父节点、右孩子节点大于父节点

试题（58）分析

本题考查数据结构的基础知识。

树的带权路径长度为树中所有叶子节点的带权路径长度之和。哈夫曼树是指权值为 w1，w2，…，w*n* 的 *n* 个叶子节点的二叉树中带权路径长度最小的二叉树。

构造最优二叉树的哈夫曼算法如下：

（1）根据给定的 *n* 个权值 {*w*1，*w*2，…，*wn*}，构成 *n* 棵二叉树的集合 F= {T1，T2，…，T*n*}，其中每棵二叉树 T*i* 中只有一个带权为 w*i* 的根节点，其左右子树均空。

（2）在 F 中选取两棵权值最小的二叉树作为左、右子树构造一棵新的二叉树，置新构造二叉树的根节点的权值为其左、右子树根节点的权值之和。

（3）从 F 中删除这两棵树，同时将新得到的二叉树加入到 F 中。

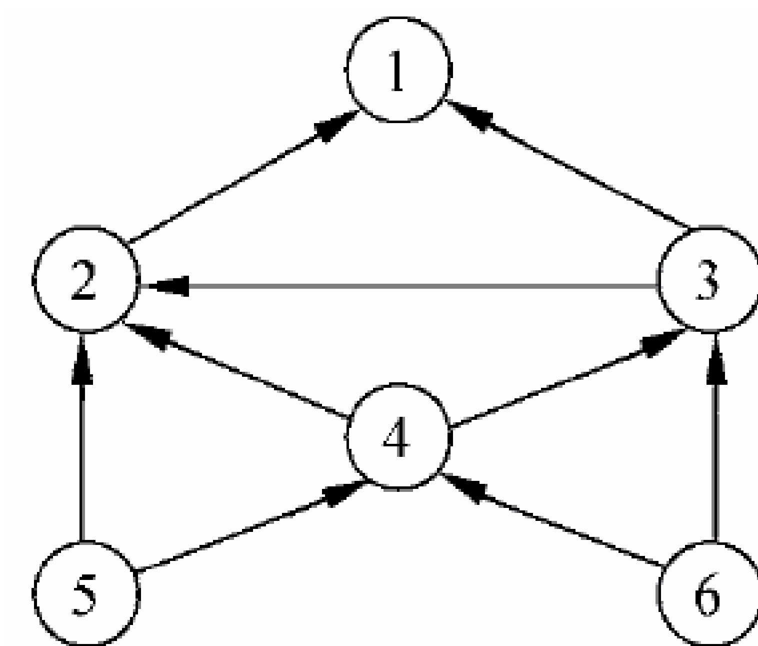
重复（2）、（3），直到 F 中只含一棵树时为止。这棵树便是最优二叉树（哈夫曼树）。

从以上叙述可知，哈夫曼树中权值最小的两个节点互为兄弟节点。

参考答案

（58）C

试题（59）



__（59）__是右图的合法拓扑序列。

（59）A.　6 5 4 3 2 1

B.　1 2 3 4 5 6

C.　5 6 3 4 2 1

D.　5 6 4 2 1 3

试题（59）分析

本题考查数据结构的基础知识。

拓扑排序是将 AOV 网中所有顶点排成一个线性序列的过程，并且该序列满足：若在 AOV 网中从顶点 v*i* 到 v*j* 有一条路径，则在该线性序列中，顶点 v*i* 必然在顶点 v*j* 之前。

对 AOV 网进行拓扑排序的方法如下：

（1）在 AOV 网中选择一个入度为零（没有前驱）的顶点且输出它；

（2）从网中删除该顶点及与该顶点有关的所有边；

（3）重复上述两步，直至网中不存在入度为零的顶点为止。

本题中只有序列“6 5 4 3 2 1”可由上述过程导出。

对有向图进行拓扑排序的结果会有两种情况：一种是所有顶点已输出，此时整个拓扑排序完成，说明网中不存在回路；另一种是尚有未输出的顶点，剩余的顶点均有前驱顶点，表明网中存在回路。

参考答案

（59）A

试题（60）

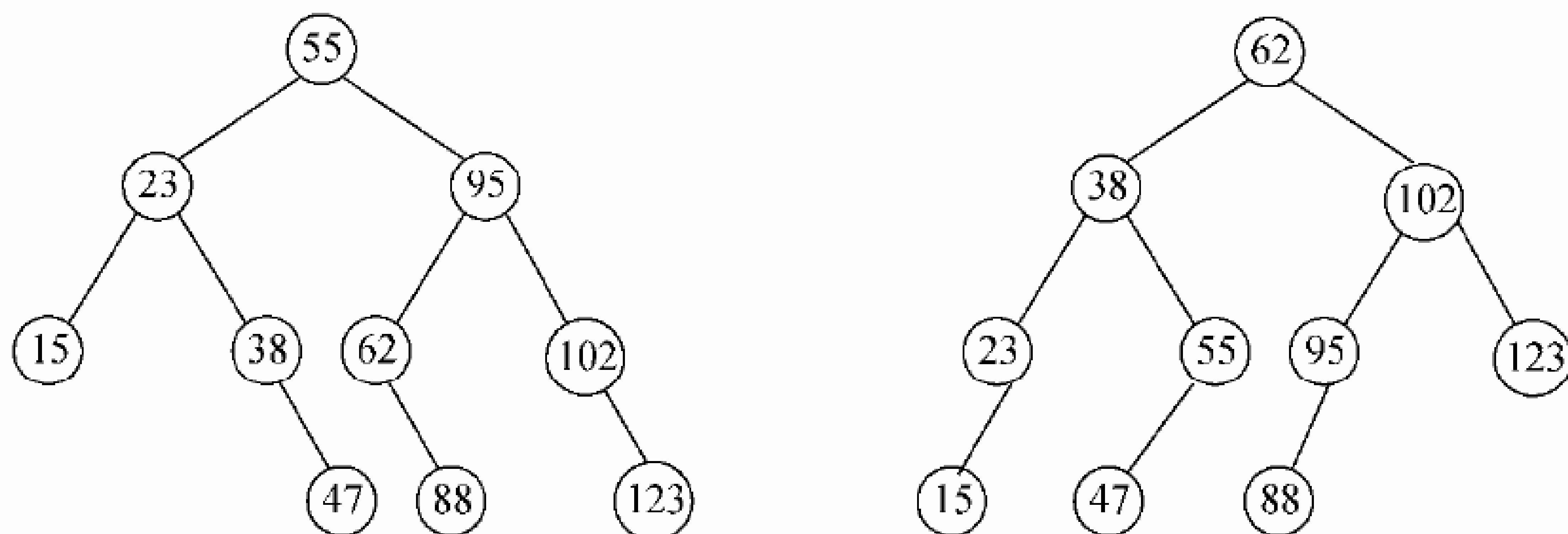
某一维数组中依次存放了数据元素 15，23，38，47，55，62，88，95，102，123，采用折半（二分）法查找元素 95 时，依次与__（60）__进行了比较。

（60）A. 62，88，95　B. 62，95　C. 55，88，95　D. 55，95

试题（60）分析

本题考查数据结构的基础知识。

对序列 15，23，38，47，55，62，88，95，102，123 进行二分查找的过程可用以下二叉树之一描述，其中，左图描述的是除 2 以后向下取整时的判定过程，右图则对应除 2 以后向上取整时的判定过程。



从上图可知，二分法查找 95 时，参与比较的元素依次为 55、95，或者 62、102、95。

参考答案

（60）D

试题（61）

已知一棵度为 3 的树（一个节点的度是指其子树的数目，树的度是指该树中所有节点的度的最大值）中有 5 个度为 1 的节点，4 个度为 2 的节点，2 个度为 3 的节点，那么，该树中的叶子节点数目为__（61）__。

（61）A. 10　B. 9　C. 8　D. 7

试题（61）分析

本题考查数据结构的基础知识。

设树中的节点总数为 n、分支数目为 m，那么 n=5+4+2+叶子节点数，$m=5\times1+4\times2+2\times3$。

在树中，节点总数等于分支数目加上 1，即 $n=m+1$。

因此，叶子节点数=$5\times1+4\times2+2\times3+1-5-4-2=9$

参考答案

（61）B

试题（62）

某算法的时间复杂度可用递归式 $T(n)=\begin{cases} O(1), & n=1 \\ 2T(n/2)+n\lg n, & n>1 \end{cases}$ 表示，若用 Θ 表示该算法的渐进时间复杂度的紧致界，则正确的是__（62）__。

（62）A. $\Theta(n\lg^2 n)$　　B. $\Theta(n\lg n)$

C. $\Theta(n^2)$　　D. $\Theta(n^3)$

试题（62）分析

该题可以用主方法来求解，对该递归式，a=1，$b=2$，$n^{\log_b a}=n^{\log_2 2}=n$ 而 $f(n)=n\lg n$，属于第二种情况，因此，其时间复杂度为 $\Theta(n\lg^2 n)$。该题还可以用递归树求解。

参考答案

（62）A

试题（63）

用动态规划策略求解矩阵连乘问题 $\boldsymbol{M}_1\times\boldsymbol{M}_2\times\boldsymbol{M}_3\times\boldsymbol{M}_4$，其中 $\boldsymbol{M}_1(20*5)$、$\boldsymbol{M}_2(5\times35)$、$\boldsymbol{M}_3(35\times4)$ 和 $\boldsymbol{M}_4(4\times25)$，则最优的计算次序为__（63）__。

（63）A. $((\boldsymbol{M}_1\times\boldsymbol{M}_2)\times\boldsymbol{M}_3)\times\boldsymbol{M}_4$　　B. $(\boldsymbol{M}_1\times\boldsymbol{M}_2)\times(\boldsymbol{M}_3\times\boldsymbol{M}_4)$

C. $(\boldsymbol{M}_1\times(\boldsymbol{M}_2\times\boldsymbol{M}_3))\times\boldsymbol{M}_4$　　D. $\boldsymbol{M}_1\times(\boldsymbol{M}_2\times(\boldsymbol{M}_3\times\boldsymbol{M}_4))$

试题（63）分析

矩阵连乘问题指的是确定 n 个矩阵相乘的次序，即给这 n 个相乘的矩阵加括号，使得按照该顺序进行计算时所需要的标量乘法的次数最少。用 $m[1, n]$来表示 n 个矩阵 $\boldsymbol{M}_1\times\boldsymbol{M}_2\times\cdots\times\boldsymbol{M}_n$ 相乘所需要的最小标量乘法的次数，则可以用递归式

$$m[i,j]=\begin{cases} 0 & \text{若}i=j \\ \min\limits_{i\leqslant k<j}\{m[i,k]+m[k+1,j]+p_{i-1}p_kp_j\} & \text{若}i\neq j \end{cases}$$

其中 $p_{i-1}p_kp_j$ 分别表示第 i 个矩阵的行数、第 k 个矩阵的列数和第 j 个矩阵的列数来表示该问题的最优子结构。

根据上述递归式，自底向上求解得到 $m[1,4]=3100$，对应的最优的加括号方式为 $(\boldsymbol{M}_1\times(\boldsymbol{M}_2\times\boldsymbol{M}_3))\times\boldsymbol{M}_4$。

参考答案

（63）C

试题（64）

下面 C 程序段中 count++语句执行的次数为__（64）__。

```
for(int i = 1;i <= 11;i *= 2)
```

```
        for(int j = 1; j <= i;j++)
            count++;
```

（64）A. 15　　B. 16　　C. 31　　D. 32

试题（64）分析

该题考查算法分析的基础知识，以及对算法中循环结构的掌握。

分析算法时间复杂度并不是确定算法运行的具体时间的长短，而是执行某个（某些）操作的次数。该题要求计算 count++语句执行的次数，根据上述 C 程序段可知，i=1 时执行 1 次，i=2 时执行 2 次，i=4 时执行 4 次，i=8 时执行 8 次，总共执行次数为 1+2+4+8=15 次。

参考答案

（64）A

试题（65）

__（65）__不能保证求得 0-1 背包问题的最优解。

（65）A. 分支限界法　　B. 贪心算法　　C. 回溯法　　D. 动态规划策略

试题（65）分析

0-1 背包问题是一个经典的最优化问题，问题描述为：有 n 个物品，第 i 个物品价值为 v_i，重量为 w_i，其中 v_i 和 w_i 均为非负数，背包的容量为 W，W 为非负数。现需要考虑如何选择装入背包的物品，使装入背包的物品总价值最大。该问题可以形式化地描述如下：

目标函数为 $\max\sum_{i=1}^{n} v_i x_i$，约束条件为 $\sum_{i=1}^{n} w_i x_i \leqslant W$，$x_i \in \{0,1\}$

0-1 背包问题具有最优子结构。考虑重量至多为 W 的背包中的物品的最大价值，若把第 j 个物品从背包中去掉，则剩下的背包中的物品的价值应该是 $n-1$ 项物品（除了第 j 项）背包容量为 $W-w_j$ 的子问题的最大价值。该问题可以通过动态规划算法来求得最优解。同时可以通过回溯法和分支限界法进行系统的搜索得到最优解。但是由于问题不具有贪心选择性质，即通过局部最优选择不能得到全局最优解，因此用贪心算法求解不能保证得到最优解。

参考答案

（65）B

试题（66）、（67）

公钥体系中，私钥用于__（66）__，公钥用于__（67）__。

（66）A. 解密和签名　　B. 加密和签名　　C. 解密和认证　　D. 加密和认证

（67）A. 解密和签名　　B. 加密和签名　　C. 解密和认证　　D. 加密和认证

试题（66）、（67）分析

本题考查公钥体系的概念和应用。

1976 年斯坦福大学的 Diffie 和 Hellman 提出了使用不同的密钥进行加密和解密的公钥加密算法。设 P 为明文，C 为密文，E 为公钥控制的加密算法，D 为私钥控制的解密算法，这些参数满足下列 3 个条件：

（1）D（E（P））=P

（2）不能由 E 导出 D

（3）选择明文攻击（选择任意明文-密文对以确定未知的密钥）不能破解 E

加密时计算 C=E（P），解密时计算 P=D（C）。加密和解密是互逆的。用公钥加密、私钥解密，可实现保密通信；用私钥加密、公钥解密，可实现数字签名。

参考答案

（66）A　　（67）D

试题（68）

HTTP 协议中，用于读取一个网页的操作方法为__（68）__。

（68）A. READ　　B. GET　　C. HEAD　　D. POST

试题（68）分析

本题考查考生对 HTTP 命令的掌握程度。

GET 是 HTTP 协议提供的少数操作方法中的一种，其含义是读一个网页。HEAD 命令用于读取网页头信息。POST 命令用于把消息加到指定的网页上。没有 READ 这一命令。

参考答案

（68）B

试题（69）

帧中继作为一种远程接入方式有许多优点，下面的选项中错误的是__（69）__。

（69）A. 帧中继比 X.25 的通信开销少，传输速度更快

B. 帧中继与 DDN 相比，能以更灵活的方式支持突发式通信

C. 帧中继比异步传输模式能提供更高的数据速率

D. 租用帧中继虚电路比租用 DDN 专线的费用低

试题（69）分析

本题考查数据交换网的基础知识。

帧中继（Frame Relay，FR）是为克服 X.25 交换网的缺陷、提高传输性能而发展起来的高速分组交换技术。帧中继网络不进行差错和流量控制，并且通过流水方式进行交换，所以比 X.25 网络的通信开销更少，传输速度更快。

帧中继提供面向连接的虚电路服务，因而比 DDN 专线更能提高通信线路利用率，用户负担的通信费用也更低廉。在帧中继网中，用户的信息速率可以在一定的范围内变

化，从而既可以适应流式业务，又可以适应突发式业务，这使得帧中继成为远程传输的理想形式。

参考答案

（69）C

试题（70）

HTML 文档中<table>标记的 align 属性用于定义__（70）__。

（70）A. 对齐方式　　B. 背景颜色
C. 边线粗细　　D. 单元格边距

试题（70）分析

本题考查 HTML 文档中<table>标记常用的属性定义。Align 用于定义文本的对齐方式。

参考答案

（70）A

试题（71）～（75）

People are indulging in an illusion whenever they find themselves explaining at a cocktail（鸡尾酒）party, say, that they are “in computers,” or “in telecommunications,” or “in electronic funds transfer”. The implication is that they are part of the high-tech world. Just between us, they usually aren’t. The researchers who made fundamental breakthroughs in those areas are in a high-tech business. The rest of us are __（71）__ of their work. We use computers and other new technology components to develop our products or to organize our affairs. Because we go about this work in teams and projects and other tightly knit working groups（紧密联系在一起的工作小组）, we are mostly in the human communication business. Our successes stem from good human interactions by all participants in the effort, and our failures stem from poor human interactions.

The main reason we tend to focus on the __（72）__ rather than the human side of the work is not because it's more __（73）__, but because it's easier to do. Getting the new disk drive installed is positively trivial compared to figuring out why Horace is in a blue funk（恐惧）or why Susan is dissatisfied with the company after only a few months. Human interactions are complicated and never very crisp（干脆的，干净利落的）and clean in their effects, but they matter more than any other aspect of the work.

If you find yourself concentrating on the __（74）__ rather than the __（75）__, you're like the vaudeville character（杂耍人物）who loses his keys on a dark street and looks for them on the adjacent street because, as he explains, “The light is better there!”.

（71）A. creators　　B. innovators
C. appliers　　D. inventors

（72）A. technical　　B. classical
　　C. social　　D. societal

（73）A. trivial　　B. crucial
　　C. minor　　D. insignificant

（74）A. technology　　B. sociology
　　C. physiology　　D. astronomy

（75）A. technology　　B. sociology
　　C. physiology　　D. astronomy

参考译文

无论何时当人们发现自己在鸡尾酒会上向别人解释，比方说他们“在计算机领域”或“在远程通信领域”或“在电子基金转账领域”工作时，他们都会沉浸在高科技的幻觉中，这就暗示他们是高科技王国里的一分子。在我们看来，他们一般都不是。在这些领域中，只有那些有根本性突破的研究人员是在做高科技业务，我们所有其他局外人只是他们工作成果的应用者。我们用计算机和其他新技术组件来开发产品或者组织我们的事务，因为是以团队和项目以及其他紧密结合的工作小组的形式来从事这项工作的，主要在从事人类交流的业务。我们的成功源自良好的、与所有此项工作的参与者之间的人际交往，同样我们的失败原因也是由于糟糕的人际交往。

我们倾向于集中精力做技术方面，而不是人际关系方面工作的主要原因，不是因为它更重要，而是因为它更容易做。与弄清楚贺瑞斯为什么恐惧不安，或者苏珊为什么在公司只工作了几个月就对公司不满意之类的事情相比，安装一个新的磁盘驱动器肯定是微不足道的。人际交往是很复杂的，并且就效果而言从来都不会是很明晰和清楚的，但是它们比工作的任何其他方面更重要。

如果你发现自己关注的是技术而不是社会方面的问题，你就相当于在一条黑暗的街上丢失了钥匙，却到邻近的另一条街上去寻找。因为“这条街上的灯比那条街上的灯要亮一些”。

参考答案

（71）C　　（72）A　　（73）B　　（74）A　　（75）B

第 8 章　2010 下半年软件设计师下午试题分析与解答

试题一（共 15 分）

阅读以下说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某时装邮购提供商拟开发订单处理系统，用于处理客户通过电话、传真、邮件或 Web 站点所下订单。其主要功能如下：

（1）增加客户记录。将新客户信息添加到客户文件，并分配一个客户号以备后续使用。

（2）查询商品信息。接收客户提交的商品信息请求，从商品文件中查询商品的价格和可订购数量等商品信息，返回给客户。

（3）增加订单记录。根据客户的订购请求及该客户记录的相关信息，产生订单并添加到订单文件中。

（4）产生配货单。根据订单记录产生配货单，并将配货单发送给仓库进行备货；备好货后，发送备货就绪通知。如果现货不足，则需向供应商订货。

（5）准备发货单。从订单文件中获取订单记录，从客户文件中获取客户记录，并产生发货单。

（6）发货。当收到仓库发送的备货就绪通知后，根据发货单给客户发货；产生装运单并发送给客户。

（7）创建客户账单。根据订单文件中的订单记录和客户文件中的客户记录，产生并发送客户账单，同时更新商品文件中的商品数量和订单文件中的订单状态。

（8）产生应收账户。根据客户记录和订单文件中的订单信息，产生并发送给财务部门应收账户报表。

现采用结构化方法对订单处理系统进行分析与设计，获得如图 1-1 所示的顶层数据流图和图 1-2 所示的 0 层数据流图。

【问题 1】（3 分）

使用说明中的词语，给出图 1-1 中的实体 E1～E3 的名称。

【问题 2】（3 分）

使用说明中的词语，给出图 1-2 中的数据存储 D1～D3 的名称。

【问题 3】（9 分）

（1）给出图 1-2 中处理（加工）P1 和 P2 的名称及其相应的输入输出流。

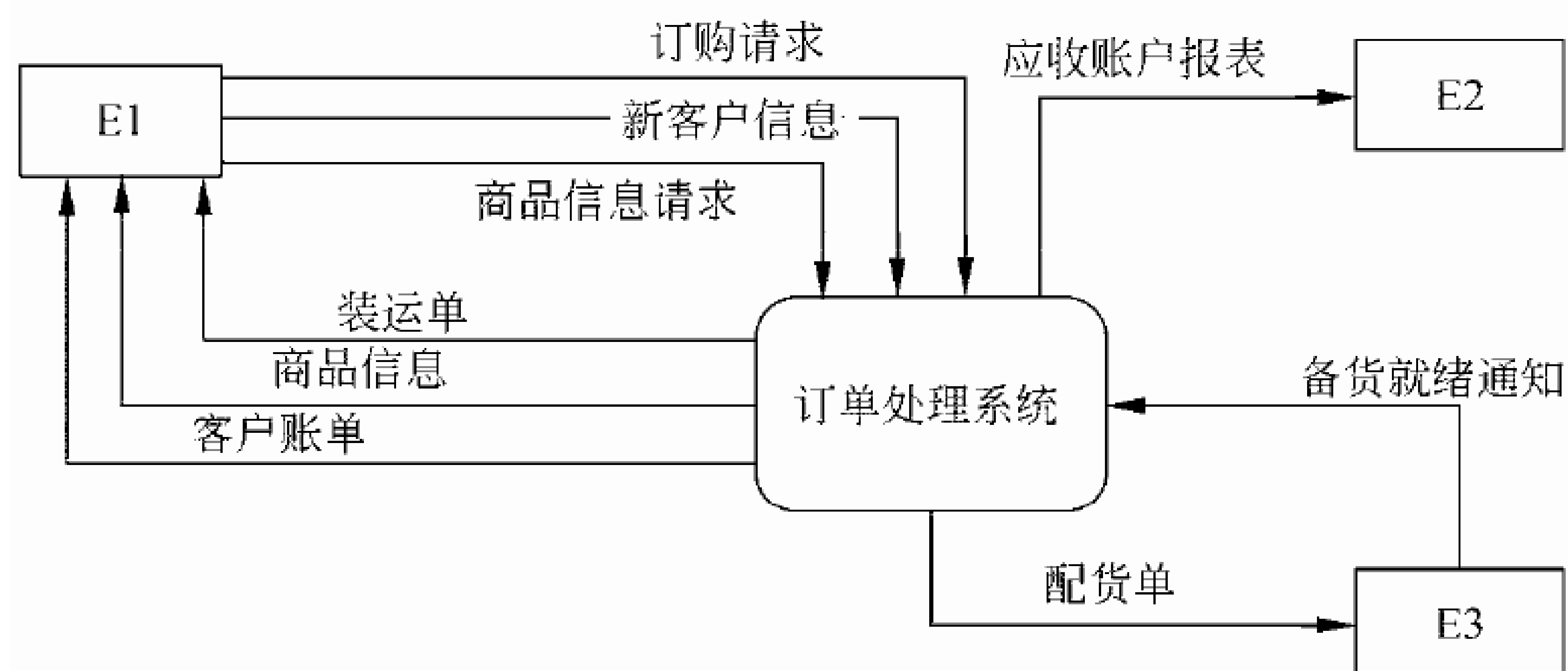


图 1-1　顶层数据流图

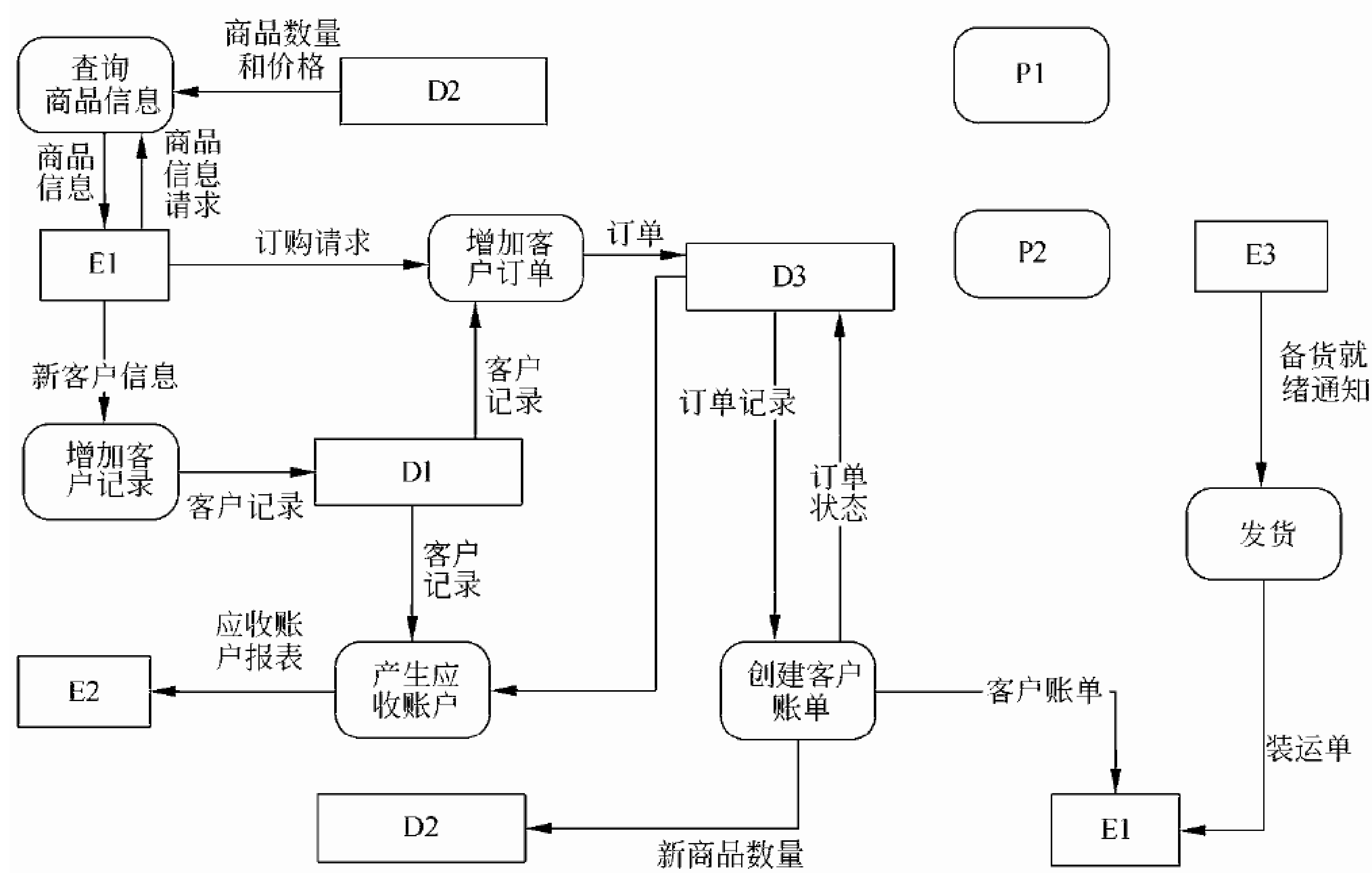


图 1-2　0 层数据流图

（2）除加工 P1 和 P2 的输入输出流外，图 1-2 还缺失了 1 条数据流，请给出其起点和终点。

起　　点	终　　点

注：名称使用说明中的词汇，起点和终点均使用图 1-2 中的符号或词汇。

试题一分析

本题考查采用结构化方法进行系统分析与设计的知识，是比较传统的题目，要求考生细心分析题目中所描述的内容。

数据流图（DFD）是一种便于用户理解、分析系统数据流程的图形化建模工具，是系统逻辑模型的重要组成部分。

【问题 1】

本问题考查顶层 DFD。顶层 DFD 一般用来确定系统边界，将待开发系统看作一个加工，因此图中只有唯一的一个处理和一些外部实体，以及这两者之间的输入输出数据流。题目要求根据描述确定图中的外部实体。根据题目中的描述，并结合已经在顶层数据流图中给出的数据流进行分析。从题目的说明中可以看出：客户提交商品信息请求、订购请求等；将配货单发送给仓库、仓库向系统发送备货就绪通知；发送给财务部门应收账户报表。由此可知该订单系统有客户、仓库和财务部门三个外部实体。对应图 1-1 中数据流和实体的对应关系，可知 E1 为客户，E2 为财务部门，E3 为仓库。本题中需注意说明（4）中向供应商订货是系统外部的行为，因此，供应商并非本系统的外部实体。

【问题 2】

本问题考查 0 层 DFD 中数据存储的确定。根据说明中的以下描述：将新客户信息添加到客户文件；从商品文件中查询商品的价格和可订购数量等商品信息；产生订单并添加到订单文件中，得出数据存储为客户文件、商品文件以及订单文件，再根据图 1-2 中 D1 的输入和输出数据流均为客户记录，D2 的输入数据流为从处理“创建客户账单”来的新商品数量，输出数据流为到处理“查询商品信息”的商品数量和价格，D3 的输入数据流为从处理“增加客户订单”来的订单，可知，D1 为客户文件，D2 为商品文件，D3 为订单文件。

【问题 3】

本问题考查 0 层 DFD 中缺失的处理和数据流。从说明中的描述功能和图 1-2，可知产生配货单和准备发货单没有在图 1-2 中，即缺少两个处理：产生配货单和准备发货单。根据说明（4）中的描述：根据订单记录产生配货单，并将配货单发送给仓库进行备货；备好货后，发送备货就绪通知。可知，产生配货单的输入流为订单记录，该输入流的起点为订单文件（D3），输出流为配货单，其终点为仓库（E3）。根据说明（5）中的描述：从订单文件中获取订单记录，从客户文件中获取客户记录，并产生发货单。可知，准备发货单的输入流为订单记录和客户记录，订单记录的起点为订单文件，客户记录的起点为客户文件；输出流为发货单。再根据说明（6）中处理发货的描述：根据发货单给客户发货，发货单的终点为处理发货。产生配货单和准备发货单分别对应 P1 和 P2（或 P2 和 P1）。

P1 和 P2 及其输入输出流均识别出来之后，再对照说明和图 1-2，以找出缺少的另外

一条数据流。对照说明（7）中的描述：根据订单文件中的订单记录和客户文件中的客户记录，产生并发送客户账单。因此，创建客户账单缺少一条输入流：客户记录，其起点为客户文件（D1）。

参考答案

【问题 1】

E1：客户　　E2：财务部门　　E3：仓库

【问题 2】

D1：客户文件　　D2：商品文件　　D3：订单文件

【问题 3】

（1）处理（加工）名称，数据流。

P1: 产生配货单　　P2: 准备发货单

数据流名称	起　点	终　点
订单记录	D3 或 订单文件	P1 或 产生配货单
配货单	P1 或 产生配货单	E3 或 仓库
订单记录	D3 或 订单文件	P2 或 准备发货单
客户记录	D1 或 客户文件	P2 或 准备发货单
发货单	P2 或 准备发货单	发货

上表中各行次序无关，但每条数据流的名称、起点、终点必须相对应。

P1 和 P2 可互换，即 P1 为“准备发货单”、P2 为“产生配货单”。

（2）缺少的数据流：

起　点	终　点
D1 或 客户文件	创建客户账单

试题二（共 15 分）

阅读以下说明，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某公司拟开发一套小区物业收费管理系统。初步的需求分析结果如下：

（1）业主信息主要包括：业主编号、姓名、房号、房屋面积、工作单位、联系电话等。房号可唯一标识一条业主信息，且一个房号仅对应一套房屋；一个业主可以有一套或多套的房屋。

（2）部门信息主要包括：部门号、部门名称、部门负责人、部门电话等。一个员工只能属于一个部门，一个部门只有一位负责人。

（3）员工信息主要包括：员工号、姓名、出生年月、性别、住址、联系电话、所在部门号、职务和密码等。根据职务不同，员工可以有不同的权限：职务为“经理”的员

工具有更改（添加、删除和修改）员工表中本部门员工信息的操作权限；职务为“收费”的员工只具有收费的操作权限。

（4）收费信息包括：房号、业主编号、收费日期、收费类型、数量、收费金额、员工号等。收费类型包括物业费、卫生费、水费和电费，并按月收取，收费标准如表2-1所示。其中：物业费=房屋面积（平方米）×每平方米单价，卫生费=套房数量（套）×每套房单价，水费=用水数量（吨）×每吨水单价，电费=用电数量（度）×每度电单价。

（5）收费完毕应为业主生成收费单，收费单示例如表2-2所示。

表2-1 收费标准

收费类型	单位	单价
物业费	平方米	1.00
卫生费	套	10.00
水　费	吨	0.70
电　费	度	0.80

表2-2 收费单示例

房号：A1608　　业主姓名：李斌

序号	收费类型	数量	金额
1	物业费	98.6	98.60
2	卫生费	1	10.00
3	水　费	6	4.20
4	电　费	102	81.60
合计	壹佰玖拾肆元肆角整		194.40

收费日期：2010-9-2　　员工号：001

【概念模型设计】

根据需求阶段收集的信息，设计的实体联系图（不完整）如图2-1所示。图2-1中收费员和经理是员工的子实体。

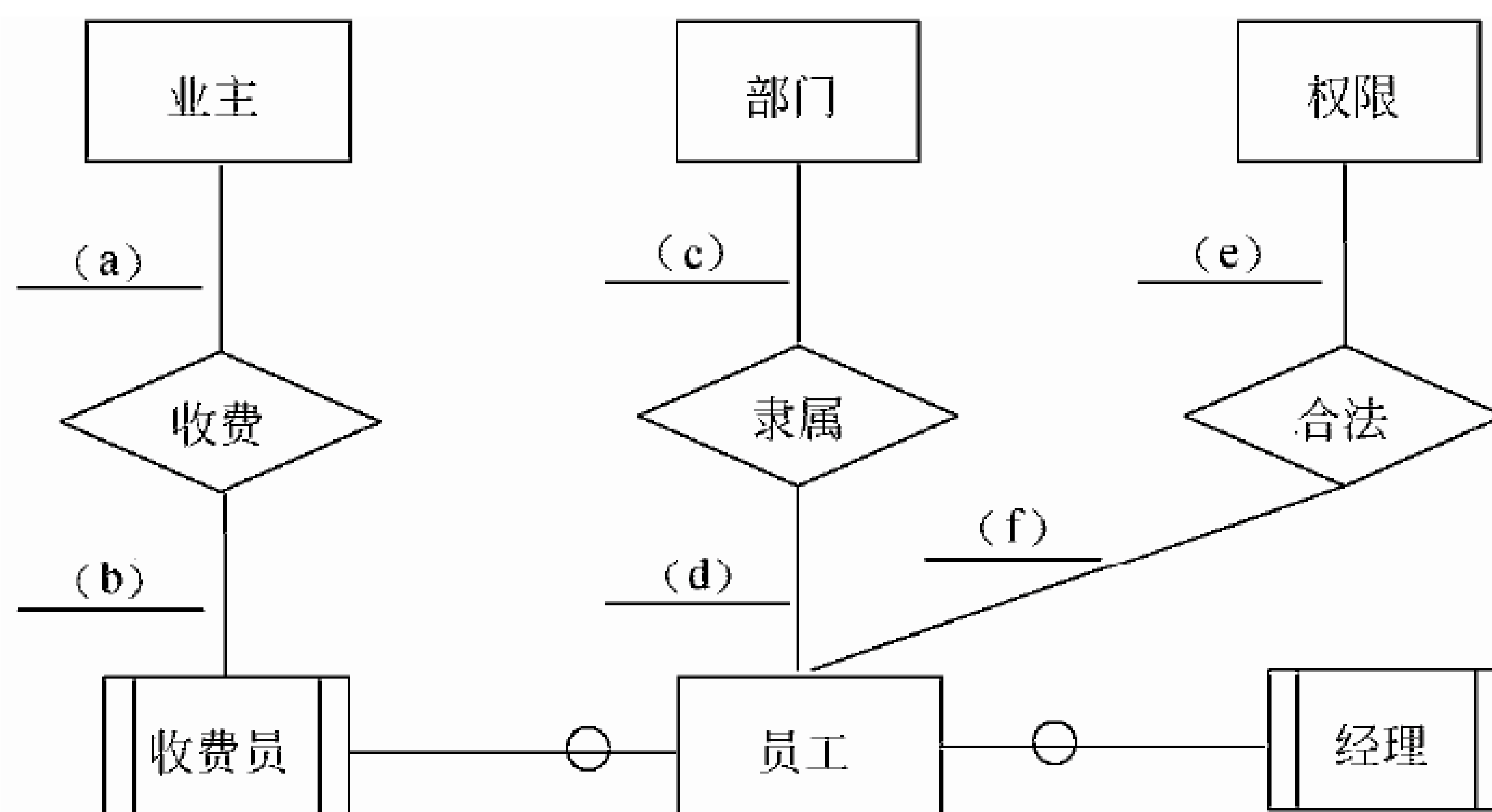


图2-1 实体联系图

【逻辑结构设计】

根据概念模型设计阶段完成的实体联系图，得出如下关系模式（不完整）：

业主（ ____（1）____ ，姓名，房屋面积，工作单位，联系电话）

员工（ ____（2）____ ，姓名，出生年月，性别，住址，联系电话，职务，密码）

部门（______(3)______，部门名称，部门电话）

权限（职务，操作权限）

收费标准（______(4)______）

收费信息（______(5)______，收费类型，收费金额，员工号）

【问题 1】（8 分）

根据图 2-1，将逻辑结构设计阶段生成的关系模式中的空（1）～（5）补充完整，然后给出各关系模式的主键和外键。

【问题 2】（5 分）

填写图 2-1 中（a）～（f）处联系的类型（注：一方用 1 表示，多方用 m 或 n 或 * 表示），并补充完整图 2-1 中的实体、联系和联系的类型。

【问题 3】（2 分）

业主关系属于第几范式？请说明存在的问题。

试题二分析

【问题 1】

根据题意，业主关系中信息主要包括：业主编号、姓名、房号、房屋面积、工作单位、联系电话等，因此，空（1）应填写“业主编号，房号”。又因为房号可唯一标识一条业主信息，所以以“房号”为主键。完整的关系模式如下：

业主（业主编号，房号，姓名，房屋面积，工作单位，联系电话）

根据题意，员工信息主要包括：员工号、姓名、出生年月、性别、住址、联系电话、所在部门号、职务和密码等，因此，空（2）应填写“员工号，所在部门号”。又因为员工号可唯一标识一条员工信息，所以“员工号”为主键。根据题意，一个员工只能属于一个部门，“所在部门号”应参照部门关系的“部门号”，因此，“所在部门号”为外键。完整的关系模式如下：

员工（员工号，所在部门号，姓名，出生年月，性别，住址，联系电话，职务，密码）

部门信息主要包括：部门号、部门名称、部门负责人、部门电话等，因此，部门关系的空（3）应填写“部门号，部门负责人”，显然该关系的主键为“部门号”。又因为部门关系的“部门负责人”应参照员工关系的“员工号”，因此，“部门负责人”为外键。

根据题意分析收费标准关系的空（4）应填写“收费类型，单位，单价”，这样收费信息关系可以根据收费类型（如水费、电费或物业费）去收费标准关系中查出单价来计算收费金额。显然收费标准关系的主键为“收费类型”。

收费信息的空（5）应填写“房号，业主编号，收费日期”，由于“房号，业主编号，收费日期”能唯一确定该关系的每一个元组，故“房号，业主编号，收费日期”为关系

的主键。又由于房号、员工号分别为业主和员工关系的主键，故“房号，员工号”为收费信息关系的外键。完整的关系模式如下：

收费信息（房号，业主编号，收费日期，收费类型，收费金额，员工号）

【问题2】

根据题意，一个员工可以为多个业主收费，同样一个业主也可以有多个员工为其收费，因此业主和收费员之间的收费联系为多对多。故空（a）应填写*，空（b）应填写*。

因为一个员工只能属于一个部门，所以部门与员工之间的隶属联系是一对多的。故空（c）应填写1，空（d）应填写*。

根据题意，职务不同员工可以有不同的权限，所以权限和员工之间的合法联系是一对多。

又由于收费员收费时必需根据收费类型（如水费、电费或物业费）到收费标准关系中查出单价来计算收费金额，所以需要增加一个收费标准关系，以及收费标准到收费联系的连线。

【问题3】

由业主关系可知：房号→业主编号，业主编号→姓名，房号→姓名，所以存在传递依赖房号→姓名。故业主关系属于第2范式。业主关系存在的问题是当某业主有多套住房时，属性“业主编号，姓名，房屋面积，工作单位，联系电话”等信息在业主关系表中重复存储，存在数据冗余。

参考答案

【问题1】

（1）业主编号，房号
　　主键：房号　　　　外键：无

（2）员工号，所在部门号
　　主键：员工号　　　　外键：所在部门号

（3）部门号，部门负责人
　　主键：部门号　　　　外键：部门负责人

（4）收费类型，单位，单价
　　主键：收费类型　　　　外键：无

（5）房号，业主编号，收费日期
　　主键：房号，业主编号，收费日期　　外键：房号，员工号

【问题2】

（a）n，或m，或*

（b）n，或m，或*

（c）1

（d）n，或m，或*

（e）1

（f）n，或 m，或*

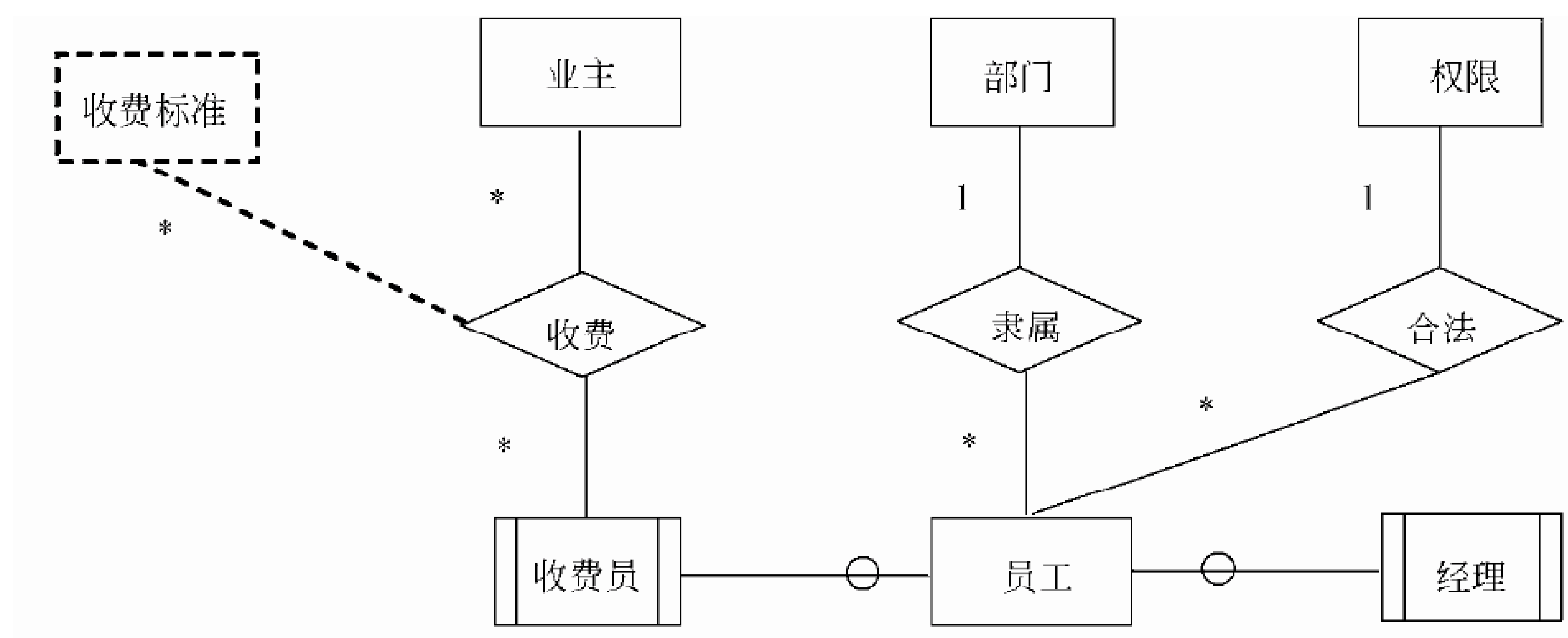


图 2-1　补充完整的实体联系图

【问题 3】

业主关系属于第 2 范式。

问题是当某业主有多套住房时，属性“业主编号，姓名，房屋面积，工作单位，联系电话”等信息在业主关系表中重复存储，存在数据冗余。

试题三（共 15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某网上药店允许顾客凭借医生开具的处方，通过网络在该药店购买处方上的药品。该网上药店的基本功能描述如下：

（1）注册。顾客在买药之前，必须先在网上药店注册。注册过程中需填写顾客资料以及付款方式（信用卡或者支付宝账户）。此外顾客必须与药店签订一份授权协议书，授权药店可以向其医生确认处方的真伪。

（2）登录。已经注册的顾客可以登录到网上药房购买药品。如果是没有注册的顾客，系统将拒绝其登录。

（3）录入及提交处方。登录成功后，顾客按照“处方录入界面”显示的信息，填写开具处方的医生的信息以及处方上的药品信息。填写完成后，提交该处方。

（4）验证处方。对于已经提交的处方（系统将其状态设置为“处方已提交”），其验证过程为：

① 核实医生信息。如果医生信息不正确，该处方的状态被设置为“医生信息无效”，并取消这个处方的购买请求；如果医生信息是正确的，系统给该医生发送处方确认请求，并将处方状态修改为“审核中”。

② 如果医生回复处方无效，系统取消处方，并将处方状态设置为“无效处方”。如果医生没有在 7 天内给出确认答复，系统也会取消处方，并将处方状态设置为“无法审核”。

③ 如果医生在 7 天内给出了确认答复，该处方的状态被修改为“准许付款”。

系统取消所有未通过验证的处方，并自动发送一封电子邮件给顾客，通知顾客处方被取消以及取消的原因。

（5）对于通过验证的处方，系统自动计算药品的价格并邮寄药品给已经付款的顾客。

该网上药店采用面向对象方法开发，使用 UML 进行建模。系统的类图如图 3-1 所示。

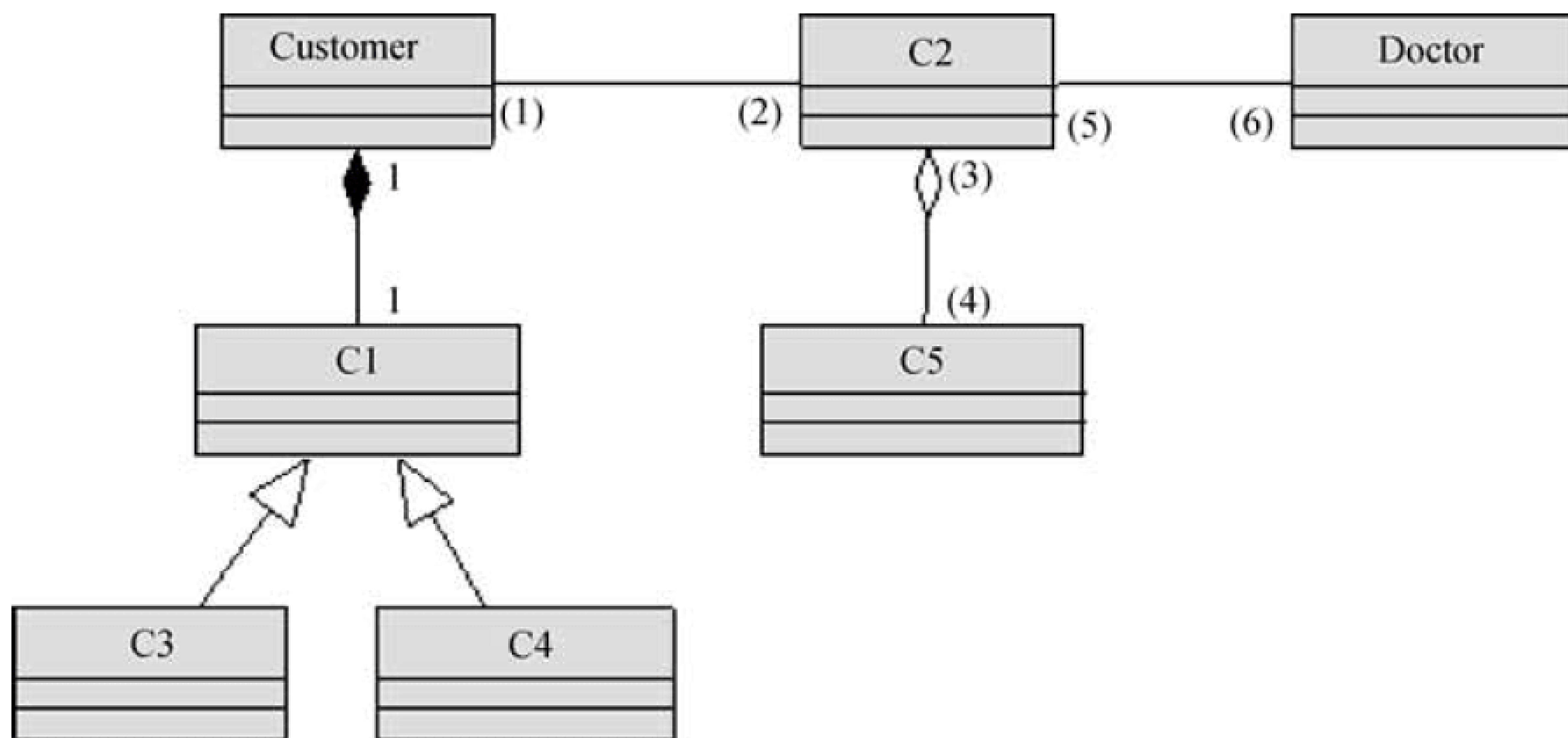


图 3-1 类图

【问题 1】（8 分）

根据说明中的描述，给出图 3-1 中缺少的 C1～C5 所对应的类名以及（1）～（6）处所对应的多重度。

【问题 2】（4 分）

图 3-2 给出了“处方”的部分状态图。根据说明中的描述，给出图 3-2 中缺少的 S1～S4 所对应的状态名以及（7）～（10）处所对应的迁移（transition）名。

【问题 3】（3 分）

图 3-1 中的符号“◆”和“◇”在 UML 中分别表示类和对象之间的哪两种关系？两者之间的区别是什么？

试题三分析

本题考查 UML 的应用，是比较传统的题目，要求考生细心分析题目中所描述的内容。

【问题 1】

本问题考查 UML 的类图。类图展现了一组对象、接口、协作和它们之间的关系。在面向对象系统的建模中，最常用的模型之一就是类图。

类图用于对系统的静态设计视图建模。这种视图主要支持系统的功能需求，即系统

要提供给用户的服务。但对系统的静态设计视图建模时，通常有三种使用方式：

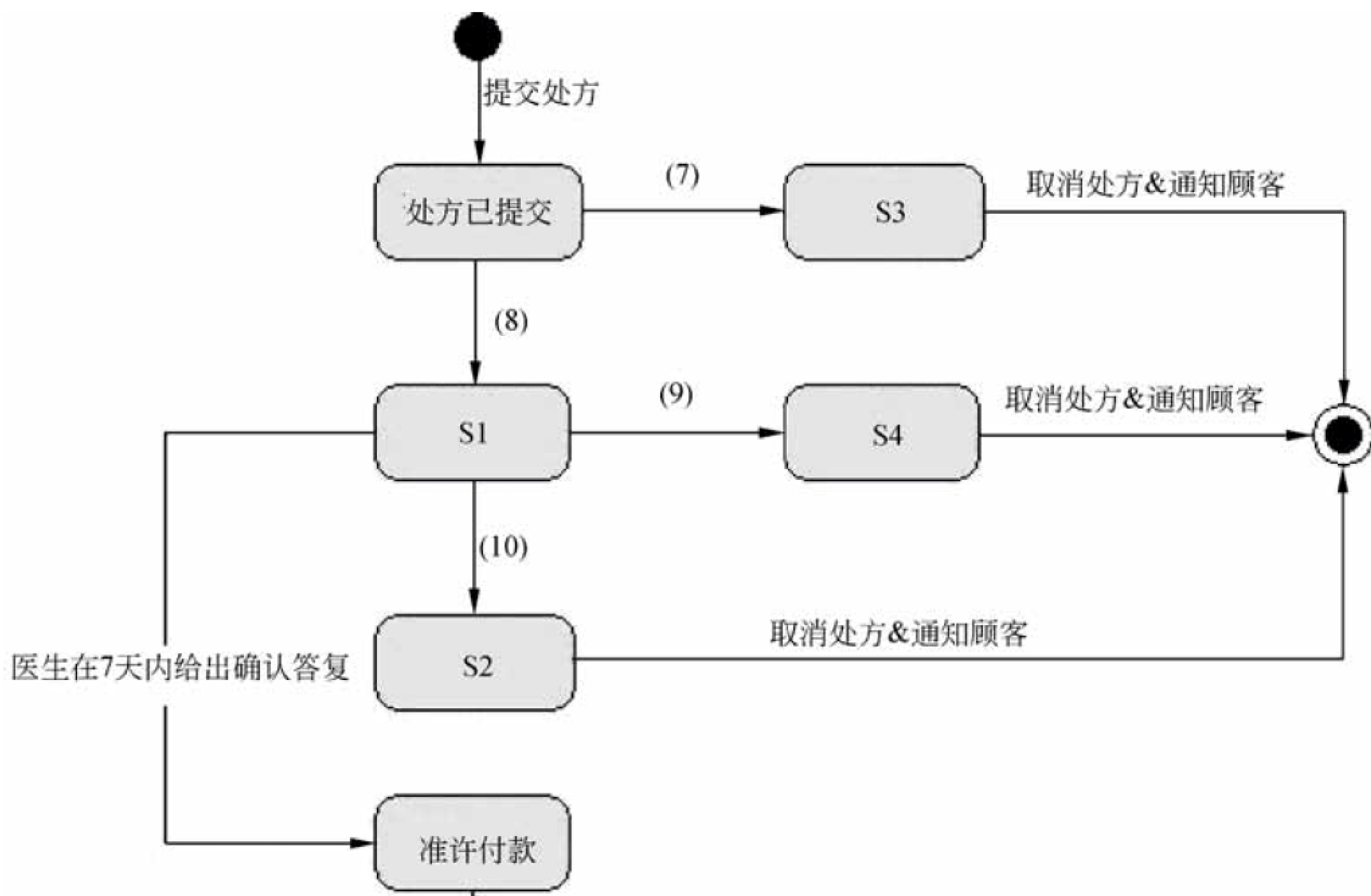


图 3-2　状态图

（1）对系统的词汇建模

对系统的词汇建模涉及做出这样的决定：哪些抽象是考虑中的系统的一部分，哪些抽象处于系统边界之外。用类图详细描述这些抽象和它们的职责。

（2）对简单的协作建模

协作是一些共同工作的类、接口和其他元素的群体，该群体提供的一些合作行为强于所有这些元素的行为之和。例如当对分布式系统的事务语义建模时，不能仅仅盯着一个单独的类来推断要发生什么，而要有相互协作的一组类来实现这些语义。用类图对这组类以及它们之间的关系进行可视化和详述。

（3）对逻辑数据库模式建模

将模式看作数据库的概念设计的蓝图。在很多领域中，要在关系数据库或面向对象数据库中存储永久信息。可以用类图对这些数据库的模式建模。

本题主要使用类图对系统词汇进行建模。题目中已经给出了类图的基本框架及部分的类，要求考生将类图中其余的类补充完整。在解答这类题目时，需要细心阅读说明中的文字，并记录和整理其中出现的名词。这些名词将来有可能成为类。其次应特别关注类图中出现的特殊关联关系，如继承关系、聚集/组装关系等。

在本题中，首先考查类图中的 Customer、C2 和 Doctor 这三个类。由说明可知，在网上购药时，顾客与医生之间不会直接发生交互，而是通过顾客持有的“处方”而发生

关联。由此可以确定 C2 对应的类应该是“处方”。

C2 与 C5 之间是聚集关系，其中 C2 表示整体类，C5 表示部分类。由于已经确定了 C2 表示的是“处方”类，那么 C5 表示就应该是处方所包含的内容。处方中包含的是药品，所以 C5 对应的类应该是“处方上的药品”。

下面来分析类图中的继承关系。继承关系表示类之间的“一般/特殊”关系。C1 表示一般类，C3 和 C4 是 C1 的两个具体类；并且这三个类与 Customer 之间具有组装关系。那么在说明中出现的所有名词词汇中，具有明显的一般/特殊关系的就是“付款方式”、“信用卡”和“支付宝账户”。“信用卡”和“支付宝账户”是具体的付款形式，当顾客付款的时候选择二者中的一个。而且每一次付款都与一个特定的顾客（即类 Customer 的一个实例）相关，没有顾客就不会发生付款行为。所以 C1 对应的类应该是“付款方式”、C3 和 C4 分别对应的是类“信用卡”“支付宝账户”。

多重度表示一个类的实例与多少个另一个类的实例发生关联。因此，在确定多重度时需要关注说明中关于类之间关系的描述。

首先来看 C2 和 C5，这两个类之间是聚集关系。前面已经确定了 C2 和 C5 分别对应类“处方”和“处方上的药品”。一张处方上应包含 1 种或多种药品。这样很容易确定出（3）和（4）的多重度应分别为 1 以及 1..*。

“处方”和“医生”之间的关系如下：一名医生可以开多张处方，也可以不开处方，所以（5）处的多重度应该为 0..*；而一张处方必定是由一名医生开具的，所以（6）处的多重度应该为 1。

“顾客”与“处方”之间的关系如下：一个顾客可以持有多张处方来买药，也可以没有处方，这样就不会发生购买行为。所以（2）处的多重度应该为 0..*。而每张处方一定属于一名顾客，所以（1）处的多重度应该为 1。

【问题 2】

状态图关注系统的动态视图，它注重描述可能的状态序列，以及在特定状态下对象对外部离散事件的响应动作。

本题考查的是类“处方”的对象的状态变化。关于网上药店对“处方”的处理流程，在说明的（4）验证处方中，给出了详细的描述。对该描述进行分析之后，可以用下面的表来说明“处方”在整个验证流程中所经历的状态。

处 方 状 态	产生该状态的原因	验 证 结 果
医生信息无效	医生信息不正确	不通过
审核中	医生信息正确	---------------
无效处方	医生回复处方无效	不通过
无法审核	医生没有在 7 天内给出答复	不通过
准许付款	医生在 7 天内给出确认答复	通过

下一步工作就是把上表中的信息与题中的状态图对应起来。

由说明可知，处方提交后的第一步操作就是核实医生信息，而这个操作会产生两种结果：医生信息正确，或者不正确。医生信息不正确会使处方的状态变更为“医生信息无效”，并导致购买行为被取消，即表中的第一行。对于这种情况，“处方”的状态变更轨迹为：处方已提交→医生信息无效→结束。而在状态图中与这条轨迹匹配的状态序列就是：处方已提交→S3→结束。由此可以确定，S3 对应的就是状态“医生信息无效”，而（7）对应的迁移就是“医生信息不正确”。

相应地，就可以判断出（8）应该代表的是核实医生信息的另一种结果，因此（8）对应的迁移应该是“医生信息正确”。由上表可知，医生信息正确时，处方状态会变更为“审核中”，这样 S1 对应的状态就是“审核中”。

但处方在状态“审核中”时，实际上会有三个后续状态：一个是图中已经给出的“准许付款”，另外两个是“无效处方”和“无法审核”。而产生这两个状态的原因分别是“医生回复处方无效”和“医生没有在 7 天内给出答复”。由此得出，（9）对应“医生回复处方无效”，S4 对应状态“无效处方”；（10）对应“医生没有在 7 天内给出答复”，S2 对应“无法审核”。

如果 S2 为状态“无效处方”，那么（10）就对应着“医生回复处方无效”；S4 对应状态“无法审核”，那么（9）就对应着“医生没有在 7 天内给出答复”。

【问题 3】

在 UML 中，◆和◇分别表示组合（composition）和聚合（aggregation）。两者的主要区别是，在组合关系中，整体对象与部分对象具有同一的生存周期。当整体对象不存在时，部分对象也不存在。而在聚合关系中，对整体对象与部分对象没有这样的要求。

参考答案

【问题 1】

C1：付款方式　　C2：处方　　C3：信用卡　　C4：支付宝账户

C5：处方上的药品（或药品）　　（C3，C4 可以互换）

（1）1　　（2）0..*　　（3）1

（4）1..*　　（5）0..*　　（6）1

【问题 2】

S1：审核中　S2：无法审核　S3：医生信息无效　S4：无效处方

（7）医生信息不正确　　（8）医生信息正确

（9）医生回复处方无效　　（10）医生没有在 7 天内给出确认答复

或者：

S2：无效处方　　S4：无法审核

（9）医生没有在 7 天内给出确认答复　　（10）医生回复处方无效

S1、S3、（7）、（8）同上

【问题 3】

◆表示组合（composition），◇表示聚合（aggregation）。

在组合关系中，整体对象与部分对象具有同一的生存周期。当整体对象不存在时，部分对象也不存在。（1 分）

而在聚合关系中，对整体对象与部分对象没有这样的要求。

试题四（共 15 分）

阅读下列说明和 C 代码，回答问题 1 至问题 3，将解答写在答题纸的对应栏内。

【说明】

堆数据结构定义如下：

对于 n 个元素的关键字序列 $\{a_1,a_2,\cdots,a_n\}$，当且仅当满足下列关系时称其为堆。

$$\begin{cases} a_i \leqslant a_{2i} \\ a_i \leqslant a_{2i+1} \end{cases} \quad 或 \quad \begin{cases} a_i \geqslant a_{2i} \\ a_i \geqslant a_{2i+1} \end{cases} \quad 其中，\ i=1,2,\cdots,\left\lfloor n/2 \right\rfloor$$

在一个堆中，若堆顶元素为最大元素，则称为大顶堆；若堆顶元素为最小元素，则称为小顶堆。堆常用完全二叉树表示，图 4-1 是一个大顶堆的例子。

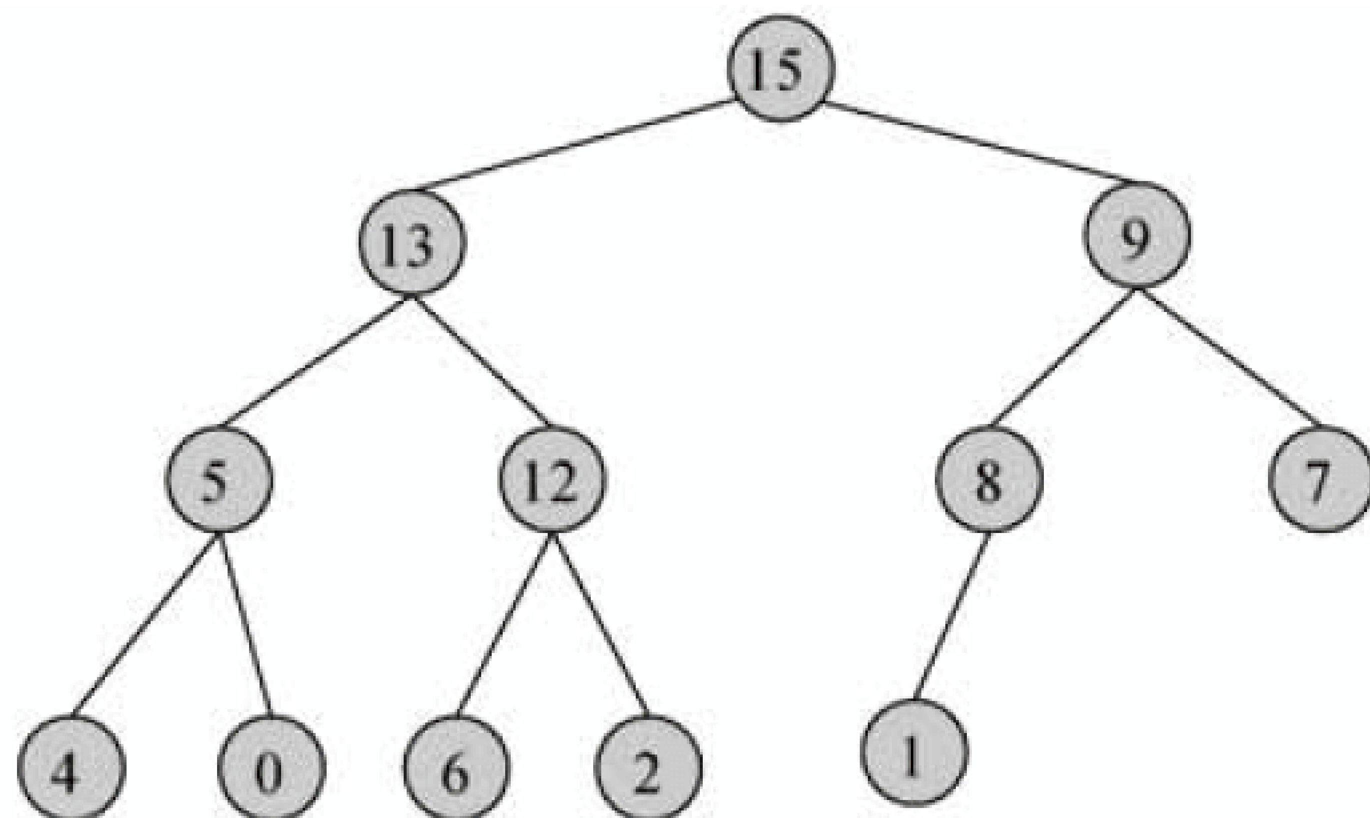


图 4-1　大顶堆示例

堆数据结构常用于优先队列中，以维护由一组元素构成的集合。对应于两类堆结构，优先队列也有最大优先队列和最小优先队列，其中最大优先队列采用大顶堆，最小优先队列采用小顶堆。以下考虑最大优先队列。

假设现已建好大顶堆 A，且已经实现了调整堆的函数 heapify(A, n, index)。

下面将 C 代码中需要完善的三个函数说明如下：

（1）heapMaximum(A)：返回大顶堆 A 中的最大元素。

（2）heapExtractMax(A)：去掉并返回大顶堆 A 的最大元素，将最后一个元素“提前”到堆顶位置，并将剩余元素调整成大顶堆。

（3）maxHeapInsert(A, key)：把元素 key 插入到大顶堆 A 的最后位置，再将 A 调整成大顶堆。

优先队列采用顺序存储方式，其存储结构定义如下：

```
#define PARENT(i)  i/2
typedef struct array{
    int *int_array;  //优先队列的存储空间首地址
    int array_size;  //优先队列的长度
    int capacity;    //优先队列存储空间的容量
} ARRAY;
```

【C 代码】

（1）函数 heapMaximum

```
int heapMaximum(ARRAY *A){  return ____(1)____;   }
```

（2）函数 heapExtractMax

```
int heapExtractMax(ARRAY *A){
    int max;
    max = A->int_array[0];
    ____(2)____;
    A->array_size --;
    heapify(A,A->array_size,0); //将剩余元素调整成大顶堆
    return max;
}
```

（3）函数 maxHeapInsert

```
int maxHeapInsert(ARRAY *A,int key){
    int i,*p;
    if (A->array_size == A->capacity)   {  //存储空间的容量不够时扩充空间
        p = (int*)realloc(A->int_array, A->capacity *2 * sizeof(int));
        if (!p) return -1;
        A->int_array = p;
        A->capacity = 2 * A->capacity;
    }
    A->array_size ++;
    i = ____(3)____;
    while (i > 0 &&____(4)____){
        A->int_array[i] = A->int_array[PARENT(i)];
        i = PARENT(i);
    }
    ____(5)____;
return 0;
}
```

【问题 1】（10 分）

根据以上说明和 C 代码，填充 C 代码中的空（1）～（5）。

【问题 2】（3 分）

根据以上 C 代码，函数 heapMaximum、heapExtractMax 和 maxHeapInsert 的时间复杂度的紧致上界分别为___（6）___、___（7）___和___（8）___（用 *O* 符号表示）。

【问题 3】（2 分）

若将元素 10 插入到堆 A =〈15, 13, 9, 5, 12, 8, 7, 4, 0, 6, 2, 1〉中，调用 maxHeapInsert 函数进行操作，则新插入的元素在堆 A 中第___（9）___个位置（从 1 开始）。

试题四分析

本题考查算法设计与分析以及 C 程序设计语言的相关知识。

【问题 1】

根据题干说明，函数 heapMaximum 返回大顶堆 A 的最大元素，即堆顶元素，因此空（1）处应填 A->int_array[0]。

函数 heapExtractMax(A)取出大顶堆 A 的最大元素，将最后一个元素“提前”到堆顶位置，并将剩余元素调整成大顶堆。因此在将堆顶元素赋给 max 后，应该将堆的最后一个元素移到堆顶位置，即空（2）处应填 A->int_array[0] = A->int_array[A->array_size–1]。

函数 maxHeapInsert(A, key)把元素 key 插入到大顶堆 A 的最后位置，再将 A 调整成大顶堆。该函数前面的代码行考虑的是当存储空间不够时扩展存储空间。而后面是根据该函数的定义实现的问题求解的算法表示，A->array_size ++;表示为堆的规模增加 1，i 表示堆的最后一个元素的下标，即新插入的元素的下标，应该为 A->array_size – 1。while 循环是自下而上调整堆，当还没有到堆顶位置，且新插入的元素大于其父亲元素，即 A->int_array[PARENT(i)] <key 时，i 变为其父亲元素的下标。直到 i 到达堆顶位置，说明新插入的元素为最大值，或者 i 的父亲元素大于新插入的元素，说明新插入的元素在 i 处，因此空（5）处填 A->int_array[i] = key。

【问题 2】

本问题考查算法的时间复杂度。

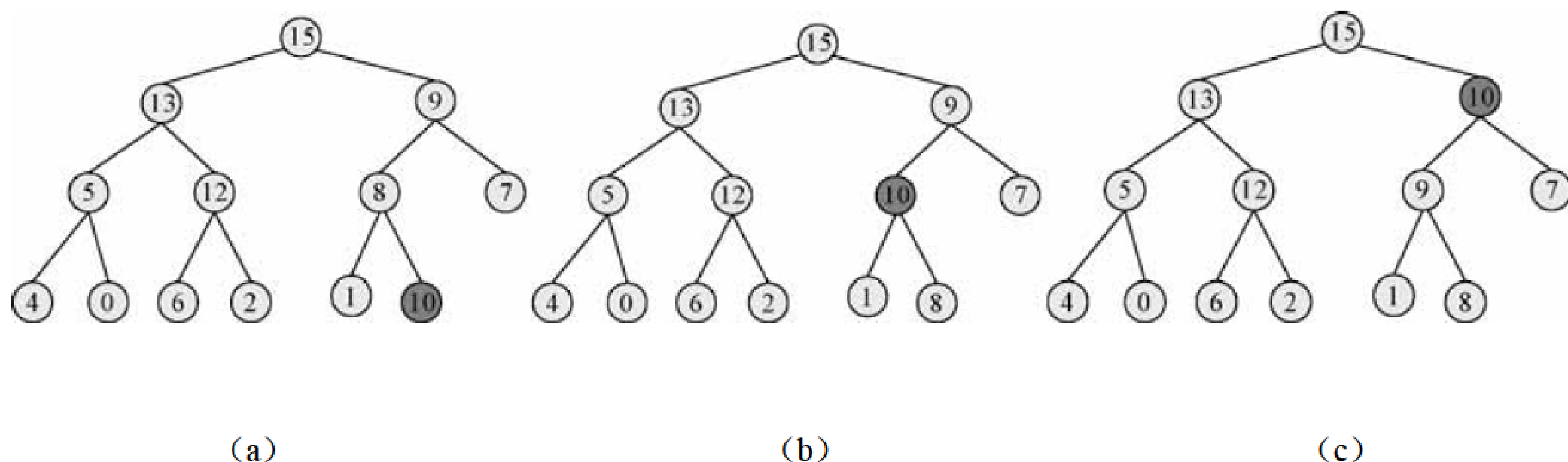
根据上述 C 代码，函数 heapMaximum 返回数组 A 的第 1 个元素，因此为常数时间即 $O(1)$。

函数 heapExtractMax 首先将数组 A 的第 1 个元素的值放到变量 max 中，然后将最后一个元素提到堆顶，最后再进行堆的调整，因此该时间复杂度实际上是调整堆的时间复杂度，即 $O(\lg n)$。

函数 maxHeapInsert 将一个元素 key 插入到堆 A 中，具体的过程为先将堆的规模增加 1，然后将元素插入到堆的最后一个位置，最后自下而上调整该元素，其时间复杂度为堆（二叉树）的高度，即 $O(\lg n)$。

【问题 3】

将元素 10 插入到堆 A =〈15，13，9，5，12，8，7，4，0，6，2，1〉中，根据 maxHeapInsert 函数进行操作，则过程如下图（a）～（c）所示。



（a）　　（b）　　（c）

新插入的元素 10 在堆 A 中处于第 3 个位置，15 和 13 分别处于第 1 和第 2 个位置。

参考答案

【问题 1】

（1）A->int_array[0]

（2）A->int_array[0] = A->int_array[A->array_size–1]

（3）A->array_size–1

（4）A->int_array[PARENT(i)] <key

（5）A->int_array[i] = key

【问题 2】

（6）$O(1)$　　（7）$O(\lg n)$　　（8）$O(\lg n)$

【问题 3】

（9）3

试题五（共 15 分）

阅读下列说明和 C++代码，将应填入<u>（n）</u>处的字句写在答题纸的对应栏内。

【说明】

某公司的组织结构图如图 5-1 所示，现采用组合（Composition）设计模式来构造该公司的组织结构，得到如图 5-2 所示的类图。

其中 Company 为抽象类，定义了在组织结构图上添加（Add）和删除（Delete）分公司/办事处或者部门的方法接口。类 ConcreteCompany 表示具体的分公司或者办事处，分公司或办事处下可以设置不同的部门。类 HRDepartment 和 FinanceDepartment 分别表示人力资源部和财务部。

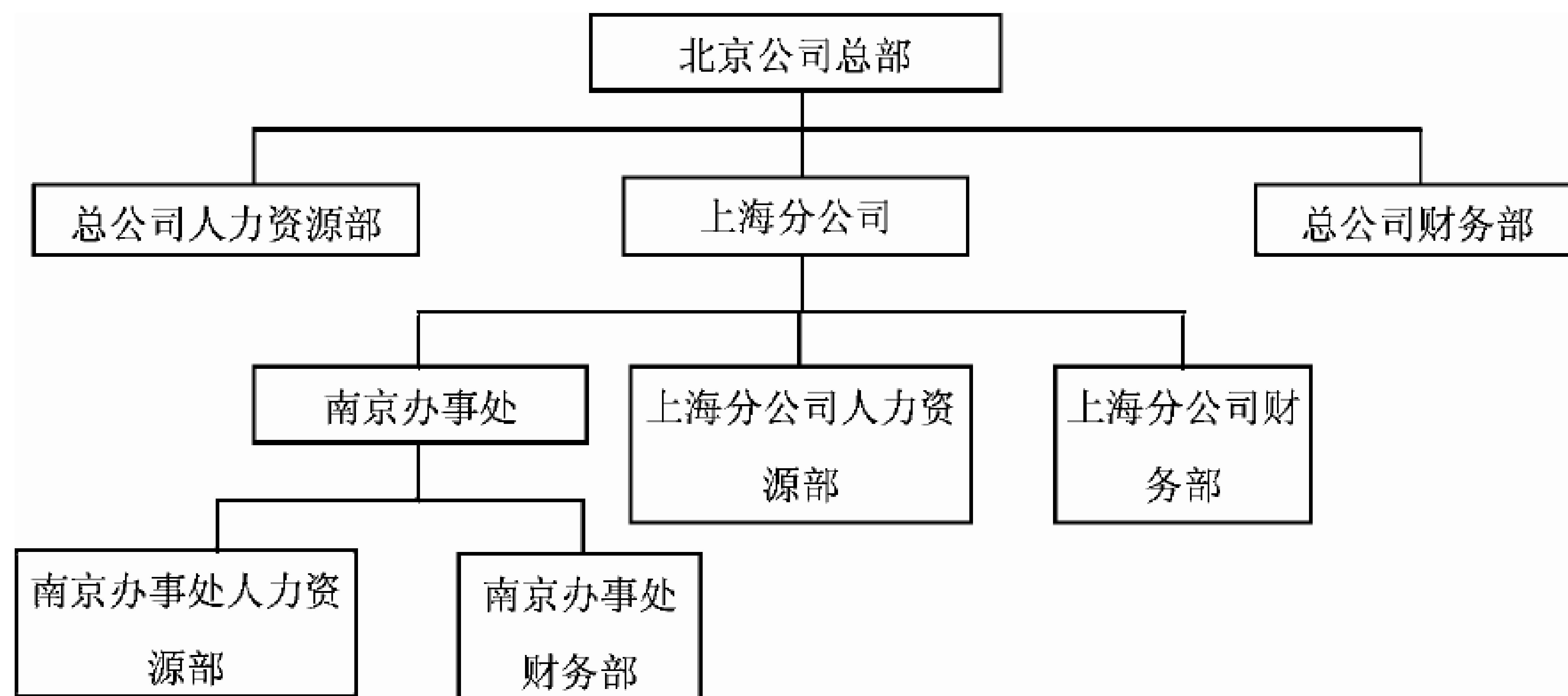


图 5-1 组织结构图

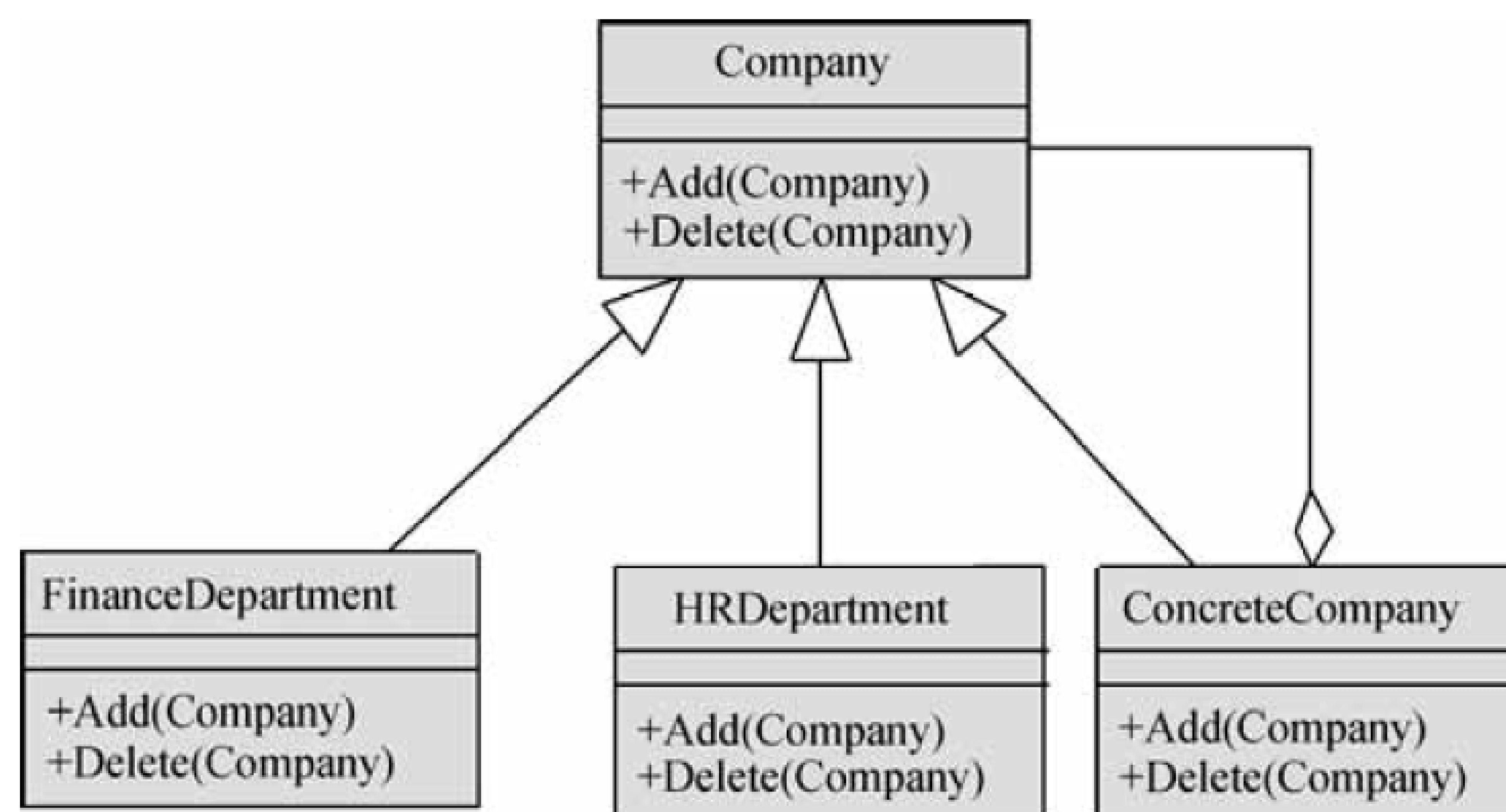


图 5-2 类图

【C++代码】

```
#include <iostream>
#include <list>
#include <string>
using namespace std;

class Company {                    // 抽象类
    protected:
        string name;
    public:
        Company(string name) { ___(1)___ = name; }
        ___(2)___;            // 增加子公司、办事处或部门
        ___(3)___;            // 删除子公司、办事处或部门
```

```
};

class ConcreteCompany : public Company {
    private:
       list< ___(4)___ > children;                    // 存储子公司、办事处或部门
    public:
       ConcreteCompany(string name) : Company(name) { }
       void Add(Company* c) { ___(5)___.push_back(c); }
       void Delete(Company* c) { ___(6)___.remove(c); }
};

class HRDepartment : public Company {
    public:
       HRDepartment(string name) : Company(name) {}       // 其他代码省略
};

class FinanceDepartment : public Company {
    public:
       FinanceDepartment(string name) : Company(name) {}  // 其他代码省略
};

void main() {
   ConcreteCompany *root = new ConcreteCompany("北京总公司");
   root->Add(new HRDepartment("总公司人力资源部"));
   root->Add(new FinanceDepartment("总公司财务部"));

   ConcreteCompany *comp = new ConcreteCompany("上海分公司");
   comp->Add(new HRDepartment("上海分公司人力资源部"));
   comp->Add(new FinanceDepartment("上海分公司财务部"));
   ___(7)___;

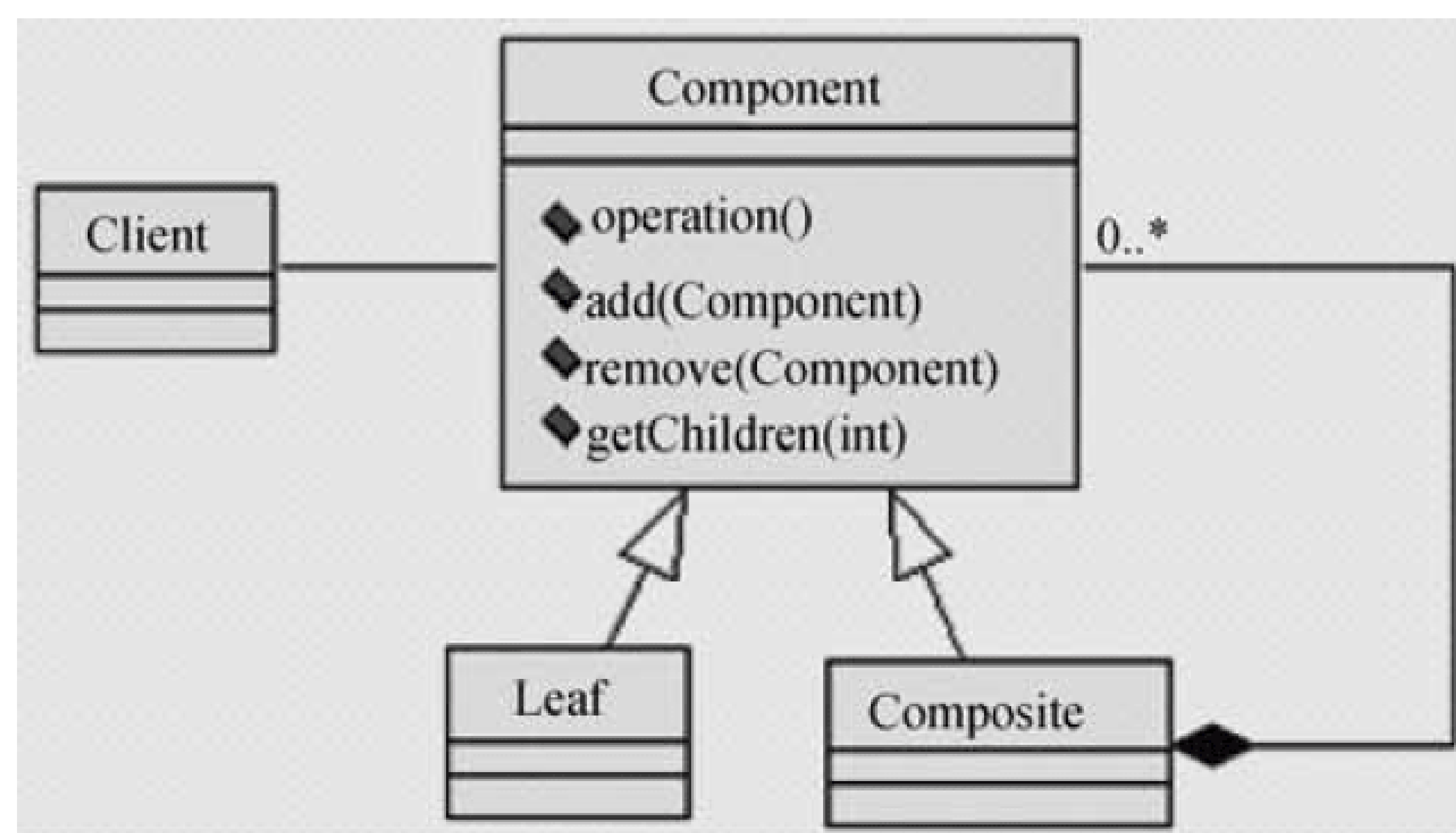
   ConcreteCompany *comp1 = new ConcreteCompany("南京办事处");
   comp1->Add(new HRDepartment("南京办事处人力资源部"));
   comp1->Add(new FinanceDepartment("南京办事处财务部"));
   ___(8)___;      //其他代码省略
}
```

试题五分析

Composite 模式将对象组合成树形结构以表示“整体-部分”的层次结构，其中的组合对象使得用户可以组合基元对象以及其他的组合对象，从而形成任意复杂的结构。

Composite 模式使得用户对单个对象和组合对象的使用具有一致性。

Composite 模式的结构如下图所示。



其中：

- 类 Component 为组合中的对象声明接口，在适当的情况下，实现所有类共有接口的缺省行为，声明一个接口用于访问和管理 Component 的子部件；
- 类 Leaf 在组合中表示叶节点对象，叶节点没有子节点；并在组合中定义图元对象的行为；
- 类 Composite 定义有子部件的那些部件的行为，存储子部件，并在 Component 接口中实现与子部件有关的操作；
- 类 Client 通过 Component 接口操纵组合部件的对象。

下列情况可以使用 Composite 模式：

（1）表示对象的整体-部分层次结构；

（2）希望用户忽略组合对象与单个对象的不同，用户将统一地使用组合结构中的所有对象。

图 5-2 中的 Company 对应的就是上图中的类 Component，ConcreteCompany 对应的是类 Composite；而上图中的 FinanceDepartment 和 HRDepartment 扮演的就是类 Leaf 的角色。

由于类 Company 的作用是为其子类提供统一的操作接口，所以将其定义为抽象类。在 C++中，抽象类的定义是：至少包含一个纯虚拟函数的类。而纯虚拟函数是没有函数体的虚拟函数，其作用是为子类提供统一接口。若要使用纯虚拟函数，必须在子类中对其进行重置。定义纯虚拟函数的语法为：

```
virtual <返回值> <函数名>(<参数列表>) = 0;
```

空（1）～（3）考查的是如何定义抽象类 Company。Company 提供了两个方法接口 Add 和 Delete，即该类中应包含两个纯虚拟函数。如何确定 Add 和 Delete 的函数原型呢？这要借助于 Company 的子类。因为子类重置父类定义的虚拟函数时，不能改变其接口定

义。所以从 ConcreteCompany 中的 Add 和 Delete 方法就能够确定出空（2）和（3）处应分别填入“virtual void Add(Company* c) = 0”和“virtual void Delete(Company* c) = 0”。空（1）考察的是在构造函数中如何给数据成员赋初值。当构造函数的参数与类的数据成员同名时，可以借助 this 指针来进行区别，因此空（1）处应填入 this->name。

空（4）～（6）考查对模式中 Composite 节点的定义。由图 5-2 可知，ConcreteCompany 与 Company 之间是聚集关系，即 ConcreteCompany 的实例中包含多个 Company 的子类的实例。为了表示这种聚集关系，使用了 C++标准类库中的类模板 list。C++的类模板必须在实例化之后才能使用。实例化类模板时，要给出类型实参。由于 children 表示的是类 Company 的子类的实例集合，所以空（4）处应填入 Company*。空（5）和（6）处分别使用了 list 中提供的方法来实现添加和删除子公司、办事处或部门。children 是 list 的实例，所以空（5）和（6）处都应填入 children。

空（7）和（8）考查的是组合模式的使用。由图 5-1 可知，组织结构图的根目录是“北京总公司”，“上海分公司”应该插入在根目录之下。所以空（7）处应填入 root->Add(comp)。而“南京办事处”是以“上海分公司”为根的子树中的节点，应插入在“上海分公司”这个节点的下面。对象 comp 表示的是以“上海分公司”为根的子树的根节点，所以空（8）处应该填入 comp->Add(comp1)。

参考答案

（1）this->name

（2）virtual void Add(Company* c) = 0

（3）virtual void Delete(Company* c) = 0

（4）Company*

（5）children

（6）children

（7）root->Add(comp)

（8）comp->Add(comp1)

试题六（共 15 分）

阅读下列说明和 Java 代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

某公司的组织结构图如图 6-1 所示，现采用组合（Composition）设计模式来设计，得到如图 6-2 所示的类图。

其中 Company 为抽象类，定义了在组织结构图上添加（Add）和删除（Delete）分公司/办事处或者部门的方法接口。类 ConcreteCompany 表示具体的分公司或者办事处，分公司或办事处下可以设置不同的部门。类 HRDepartment 和 FinanceDepartment 分别表示人力资源部和财务部。

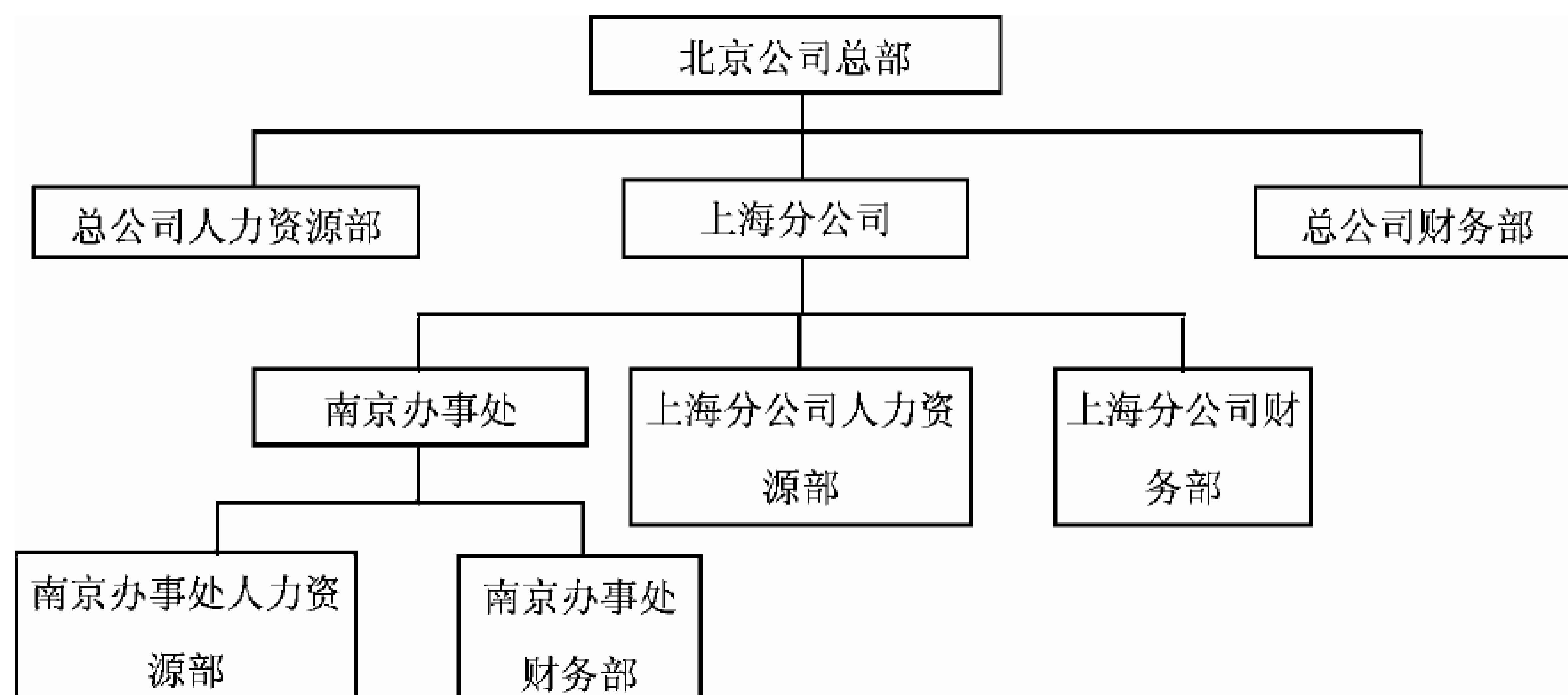


图 6-1　组织结构图

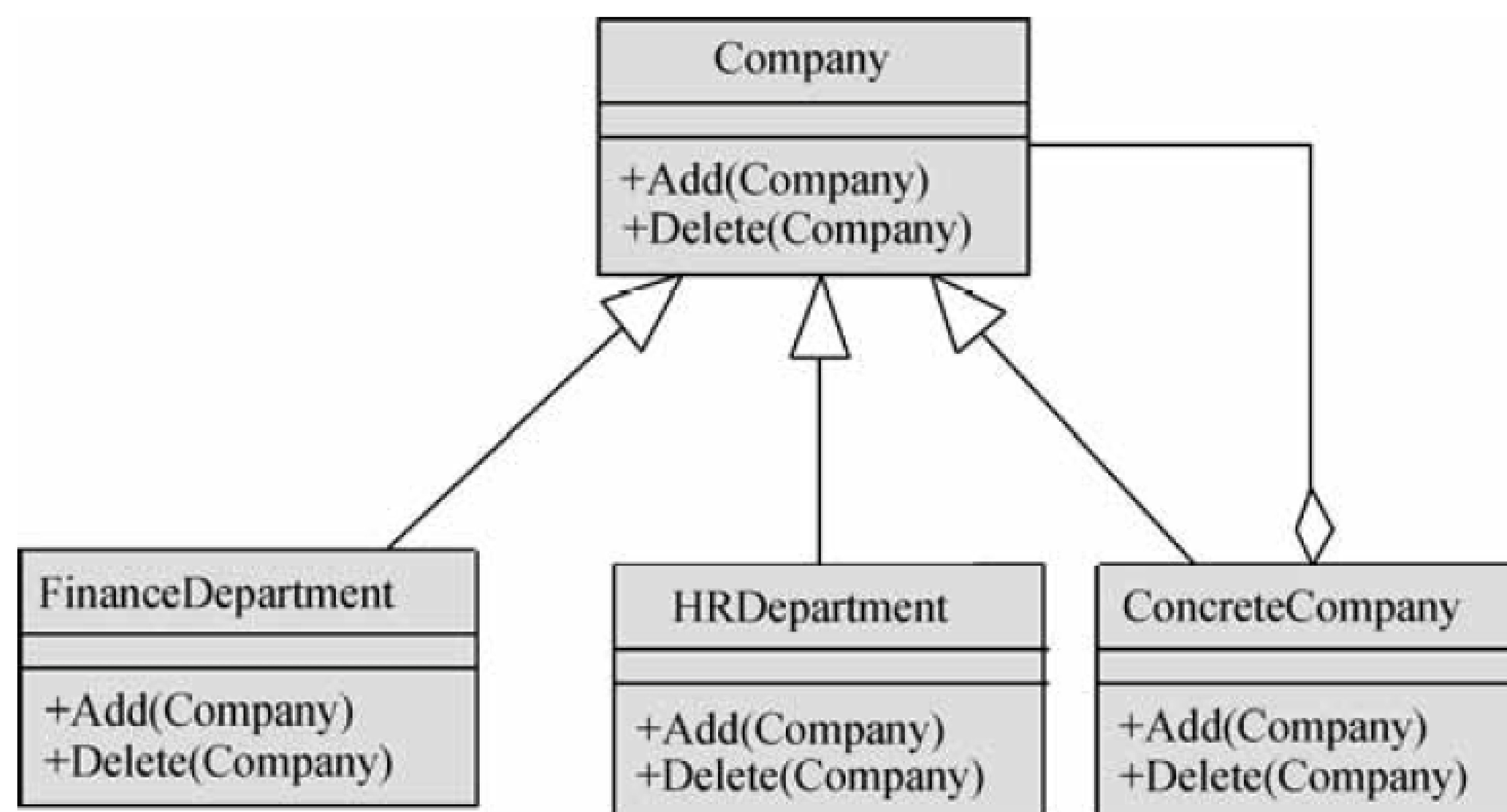


图 6-2　类图

【Java 代码】

```
import java.util.*;
___(1)___ Company {
  protected String  name;
  public Company(String name) { ___(2)___ = name; }
  public abstract void Add(Company c);       // 增加子公司、办事处或部门
  public abstract void Delete(Company c);    // 删除子公司、办事处或部门
}

class ConcreteCompany extends Company {
  private List< ___(3)___ > children = new ArrayList< ___(4)___ >();
// 存储子公司、办事处或部门
  public ConcreteCompany(String name) { super(name); }
  public void Add(Company c) { ___(5)___.add(c);   }
```

```
    public void Delete(Company c) { ___(6)___.remove(c); }
  }

  class HRDepartment extends Company {
     public HRDepartment(String name)  { super(name); }
     // 其他代码省略
  }

  class FinanceDepartment extends Company {
     public FinanceDepartment(String name)  { super(name); }
     // 其他代码省略
  }

  public class Test  {
    public static void main(String[] args) {
      ConcreteCompany root = new ConcreteCompany("北京总公司");
      root.Add(new HRDepartment("总公司人力资源部"));
      root.Add(new FinanceDepartment("总公司财务部"));

      ConcreteCompany comp = new ConcreteCompany("上海分公司");
      comp.Add(new HRDepartment("上海分公司人力资源部"));
      comp.Add(new FinanceDepartment("上海分公司财务部"));
      ___(7)___;

      ConcreteCompany comp1 = new ConcreteCompany("南京办事处");
      comp1.Add(new HRDepartment("南京办事处人力资源部"));
      comp1.Add(new FinanceDepartment("南京办事处财务部"));
      ___(8)___;       // 其他代码省略
    }
  }
```

试题六分析

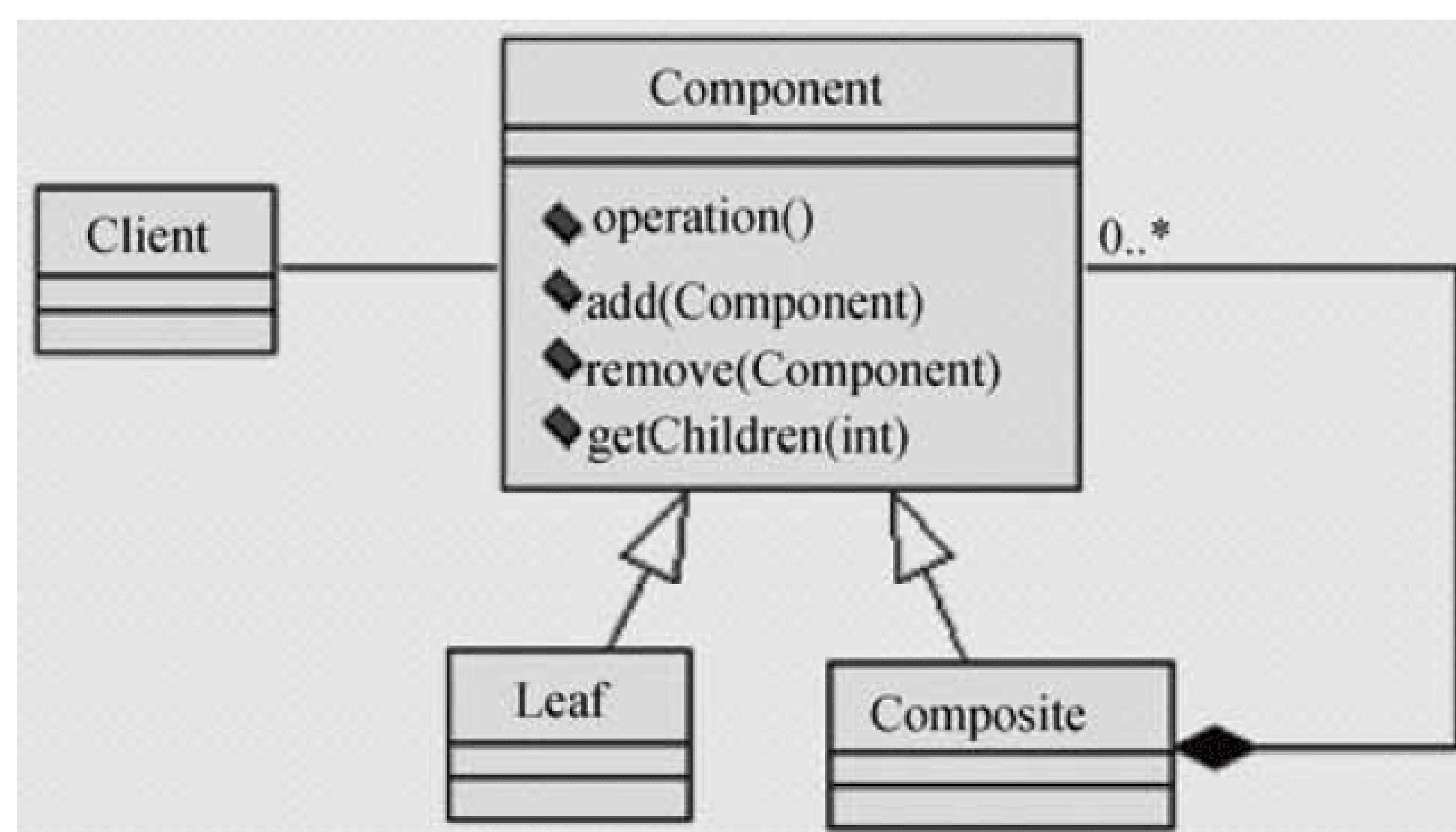
Composite 模式将对象组合成树形结构以表示“整体-部分”的层次结构，其中的组合对象使得你可以组合基元对象以及其他的组合对象，从而形成任意复杂的结构。Composite 模式使得用户对单个对象和组合对象的使用具有一致性。

Composite 模式的结构如下图所示。

其中：

- 类 Component 为组合中的对象声明接口，在适当的情况下，实现所有类共有接口的缺省行为，声明一个接口用于访问和管理 Component 的子部件；
- 类 Leaf 在组合中表示叶节点对象，叶节点没有子节点；并在组合中定义图元对

象的行为；



- 类 Composite 定义有子部件的那些部件的行为，存储子部件，并在 Component 接口中实现与子部件有关的操作；
- 类 Client 通过 Component 接口操纵组合部件的对象。

下列情况可以使用 Composite 模式：

（1）表示对象的整体-部分层次结构；

（2）希望用户忽略组合对象与单个对象的不同，用户将统一地使用组合结构中的所有对象。

图 6-2 中的 Company 对应的就是上图中的类 Component，ConcreteCompany 对应的是类 Composite；而上图中的 FinanceDepartment 和 HRDepartment 扮演的就是类 Leaf 的角色。

由于类 Company 的作用是为其子类提供统一的操作接口，所以将其定义为抽象类。空（1）～（2）考查的是如何定义抽象类 Company。在 Java 中，可以通过在类名之前加 abstract 关键字来定义抽象类，因此空（1）处应填入 abstract class。空（2）考查的是在构造函数中如何给数据成员赋初值。当构造函数的参数与类的数据成员同名时，可以借助 this 指针来进行区别，因此空（2）处应填入 this.name。

空（3）～（6）考查对模式中 Composite 节点的定义。由图 5-2 可知，ConcreteCompany 与 Company 之间是聚集关系，即 ConcreteCompany 的实例中包含多个 Company 的子类的实例。为了表示这种聚集关系，使用了 Java 包中的类模板 List。类模板必须在实例化之后才能使用。实例化类模板时，要给出类型实参。由于 children 表示的是类 Company 的子类的实例集合，所以空（3）和（4）处都应填入 Company。空（5）和（6）处分别使用了 List 中提供的方法来实现添加和删除子公司、办事处或部门。children 是 list 的实例，所以空（5）和（6）处都应填入 children。

空（7）和（8）考查的是组合模式的实用。由图 6-1 可知，组织结构图的根目录是“北京总公司”，“上海分公司”应该插入在根目录之下。所以空（7）处应填入

root.Add(comp)。而“南京办事处”是以“上海分公司”为根的子树中的节点，应插入在“上海分公司”这个节点的下面。对象 comp 表示的是以“上海分公司”为根的子树的根节点，所以空（8）处应该填入 comp.Add(comp1)。

参考答案

（1）abstract class

（2）this.name

（3）Company

（4）Company

（5）children

（6）children

（7）root.Add(comp)

（8）comp.Add(comp1)

第 9 章　2011 上半年软件设计师上午试题分析与解答

试题（1）

在 CPU 中用于跟踪指令地址的寄存器是__（1）__。

（1）A．地址寄存器（MAR）　　B．数据寄存器（MDR）
　　C．程序计数器（PC）　　D．指令寄存器（IR）

试题（1）分析

本题考查寄存器的基本知识。

CPU 中通常设置一些寄存器，用于暂时存储程序运行过程中的相关信息。其中，通用寄存器常用于暂存运算器需要的数据或运算结果，地址寄存器和数据寄存器用于访问内存时的地址和数据暂存，指令寄存器用于暂存正在执行的指令，程序计数器中存放待执行的指令的地址。

参考答案

（1）C

试题（2）

指令系统中采用不同寻址方式的目的是__（2）__。

（2）A．提高从内存获取数据的速度　　B．提高从外存获取数据的速度
　　C．降低操作码的译码难度　　D．扩大寻址空间并提高编程灵活性

试题（2）分析

本题考查指令系统的基本概念。

寻址方式是指寻找操作数或操作数地址的方式。指令系统中采用不同寻址方式的目的是为了在效率和方便性上找一个平衡。立即数寻址和寄存器寻址在效率上是最快的，但是寄存器数目少，不可能将操作数都存入其中等待使用，立即数的使用场合也非常有限，这样就需要将数据保存在内存中，然后使用直接寻址、寄存器间接寻址、寄存器相对寻址、基址加变址寻址、相对基址及变址寻址等寻址方式将内存中的数据移入寄存器中。

参考答案

（2）D

试题（3）

在计算机系统中采用总线结构，便于实现系统的积木化构造，同时可以__（3）__。

（3）A．提高数据传输速度　　B．提高数据传输量
　　C．减少信息传输线的数量　　D．减少指令系统的复杂性

试题（3）分析

本题考查计算机系统的基础知识。

总线是连接计算机有关部件的一组信号线，是计算机中用来传送信息代码的公共通道。采用总线结构主要有以下优点：简化系统结构，便于系统设计制造；大大减少了连线数目，便于布线，减小体积，提高系统的可靠性；便于接口设计，所有与总线连接的设备均采用类似的接口；便于系统的扩充、更新与灵活配置，易于实现系统的模块化；便于设备的软件设计，所有接口的软件就是对不同的口地址进行操作；便于故障诊断和维修，同时也降低了成本。

参考答案

（3）C

试题（4）

原码表示法和补码表示法是计算机中用于表示数据的两种编码方法，在计算机系统中常采用补码来表示和运算数据，原因是采用补码可以 （4） 。

（4）A．保证运算过程与手工运算方法保持一致
B．简化计算机运算部件的设计
C．提高数据的运算速度
D．提高数据的运算精度

试题（4）分析

本题考查数据表示的基础知识。

使用补码表示数据时，可以将符号位和其他位统一处理，减法也可按加法来处理，从而简化运算部件的设计。

参考答案

（4）B

试题（5）

计算机中的浮点数由三部分组成：符号位 S，指数部分 E（称为阶码）和尾数部分 M。在总长度固定的情况下，增加 E 的位数、减少 M 的位数可以 （5） 。

（5）A．扩大可表示的数的范围同时降低精度
B．扩大可表示的数的范围同时提高精度
C．减小可表示的数的范围同时降低精度
D．减小可表示的数的范围同时提高精度

试题（5）分析

本题考查数据表示的基础知识。

浮点数在计算机中用以近似表示任意某个实数，一个浮点数 a 可如下表示：

$$a = M \times b^E$$

其中，尾数部分 M 的位数越多，数的精度越高，指数部分 E 的位数越多，能表示的数值

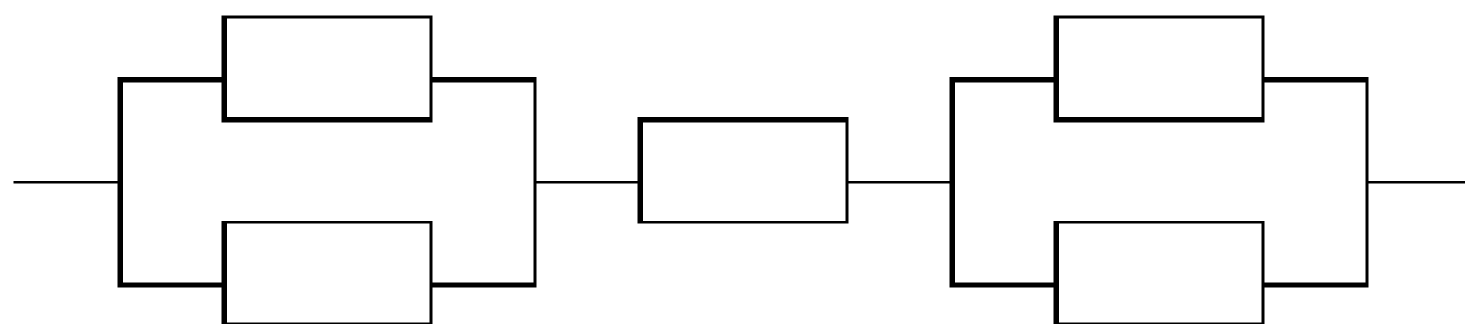
越大。因此在总长度固定的情况下，增加 E 的位数、减少 M 的位数可以扩大可表示的数的范围同时降低精度。

参考答案

（5）A

试题（6）

某计算机系统由下图所示的部件构成，假定每个部件的千小时可靠度都为 R，则该系统的千小时可靠度为__（6）__。



（6）A. $R+2R/4$　　B. $R+R^2/4$

C. $R(1-(1-R)^2)$　　D. $R(1-(1-R)^2)^2$

试题（6）分析

本题考查系统可靠性方面的基础知识。

由子系统构成串联系统时，其中任何一个子系统失效就足以使系统失效，其可靠度等于各子系统可靠度的乘积；构成并联系统时，只要有一个子系统正常工作，系统就能正常工作。

设每个子系统的可靠性分别以 $R_1, R_2, \cdots, R_N$ 表示，则整个系统用串联方式构造时的可靠度为 $R = R_1 \times R_2 \cdots \times \cdots R_N$，整个系统用串联方式构造时的可靠度为 $R = 1-(1-R_1)(1-R_2)\cdots(1-R_N)$。

因此，本系统的可靠度为 $R(1-(1-R)^2)^2$。

参考答案

（6）D

试题（7）

用户 A 从 CA 获得用户 B 的数字证书，并利用__（7）__验证数字证书的真实性。

（7）A. B 的公钥　　B. B 的私钥

C. CA 的公钥　　D. CA 的私钥

试题（7）分析

本题考查数字证书和 CA 证书授权（Certificate Authority）中心的相关知识点。

数字证书是由权威机构——CA 证书授权（Certificate Authority）中心发行的，能提供在 Internet 上进行身份验证的一种权威性电子文档，人们可以在互联网交往中用它来证明自己的身份和识别对方的身份。

数字证书包含版本、序列号、签名算法标识符、签发人姓名、有效期、主体名、主体公钥信息等并附有 CA 的签名，用户 A 获取用户 B 的数字证书后通过验证 CA 的签名

来确认数字证书的有效性。验证 CA 的签名时使用的是 CA 的公钥。

参考答案

（7）C

试题（8）

宏病毒一般感染以__（8）__为扩展名的文件。

（8）A．EXE　　B．COM　　C．DOC　　D．DLL

试题（8）分析

本题考查计算机病毒的基础知识。

病毒文件名称一般分为三部分，第一部分表示病毒的类型，如 Worm 表示蠕虫病毒，Trojan 表示特洛伊木马，Backdoor 表示后门病毒，Macro 表示宏病毒等。

宏病毒感染的对象是使用某些程序创建的文本文档、数据库、电子表格等文件。

参考答案

（8）C

试题（9）

在 IE 浏览器中，安全级别最高的区域设置是__（9）__。

（9）A．Internet　　B．本地 Intranet

C．可信站点　　D．受限站点

试题（9）分析

本题考查 IE 浏览器不同区域安全等级的基础知识。

在 IE 浏览器中，安全等级从可信站点、本地 Intranet、Internet 到受限站点默认情况下依次为低、中低、中、高，逐步提升，如下图所示。



参考答案

（9）D

试题（10）

下列关于软件著作权中翻译权的叙述不正确的是：翻译权是指__（10）__的权利。

（10）A. 将原软件从一种自然语言文字转换成另一种自然语言文字
B. 将原软件从一种程序设计语言转换成另一种程序设计语言
C. 软件著作权人对其软件享有的以其他各种语言文字形式再表现
D. 对软件的操作界面或者程序中涉及的语言文字翻译成另一种语言文字

试题（10）分析

软件著作权中翻译权是指以不同于原软件作品的一种程序语言转换该作品原使用的程序语言，而重现软件作品内容的创作的产品权利。简单地说，也就是指将原软件从一种程序语言转换成另一种程序语言的权利。

参考答案

（10）B

试题（11）

某软件公司研发的财务软件产品在行业中技术领先，具有很强的市场竞争优势。为确保其软件产品的技术领先及市场竞争优势，公司采取相应的保密措施，以防止软件技术秘密的外泄。并且，还为该软件产品冠以"用友"商标，但未进行商标注册。此情况下，公司仅享有该软件产品的__（11）__。

（11）A. 软件著作权和专利权　　B. 商业秘密权和专利权
C. 软件著作权和商业秘密权　　D. 软件著作权和商标权

试题（11）分析

由于是软件公司研发的财务软件产品，因此，软件公司享有该软件产品的软件著作权。又由于商业秘密的构成条件是：商业秘密必须具有未公开性，即不为公众所知悉；商业秘密必须具有实用性，即能为权利人带来经济效益；商业秘密必须具有保密性，即采取了保密措施。

综上所述，公司仅享有该软件产品的软件著作权和商业秘密权。

参考答案

（11）C

试题（12）

以下编码方法中，__（12）__属于熵编码。

（12）A. 哈夫曼编码　　B. 小波变换编码
C. 线性预测编码　　D. 行程编码

试题（12）分析

在计算机信息处理中，"哈夫曼编码"是一种一致性编码法（又称"熵编码法"），用于数据的无损耗压缩。这一术语是指使用一张特殊的编码表将源字符（例如某文件中的一个符号）进行编码。这张编码表的特殊之处在于，它是根据每一个源字符

出现的估算概率而建立起来的。出现概率高的字符使用较短的编码，出现概率低的则使用较长的编码，这便使编码之后的字符串的平均期望长度降低，从而达到无损压缩数据的目的。

参考答案

（12）A

试题（13）

CIF 视频格式的图像分辨率为＿（13）＿。

（13）A．352×240　　B．352×288　　C．640×480　　D．320×240

试题（13）分析

CIF 是常用的标准化图像格式（Common Intermediate Format）。在 H.323 协议簇中，规定了视频采集设备的标准采集分辨率，CIF＝352×288 像素。

参考答案

（13）B

试题（14）

由 ISO 制定的 MPEG 系列标准中，＿（14）＿是多媒体内容描述接口标准。

（14）A．MPEG-1　　B．MPEG-2　　C．MPEG-4　　D．MPEG-7

试题（14）分析

由 ISO 制定的 MPEG 系列标准中，MPEG-7 称为“多媒体内容描述接口”（multimedia content description interface）。该标准是建立对多媒体内容的描述标准，满足包括静止图像、图形、3D 模型、音频、话音、视频以及以上元素组合在一起的合成多媒体信息的应用领域的要求，并兼顾标准的通用性和扩展性的要求。

参考答案

（14）D

试题（15）

包含 8 个成员的开发小组的沟通路径最多有＿（15）＿条。

（15）A．28　　B．32　　C．56　　D．64

试题（15）分析

本题考查项目管理及工具技术。

软件开发小组的沟通路径受到小组组织形式和规模的影响。若任意小组成员之间均可能有沟通路径，则可用完全连通图来对开发小组的沟通路径建模，最多的沟通路径为完全连通图的边数，即 n 个成员的开发小组的沟通路径是 $n(n-1)/2$，因此 8 个成员的开发小组的沟通路径有 28 条。

参考答案

（15）A

试题（16）

模块 A 直接访问模块 B 的内部数据，则模块 A 和模块 B 的耦合类型为 （16） 。

（16）A. 数据耦合 B. 标记耦合 C. 公共耦合 D. 内容耦合

试题（16）分析

本题考查软件的分析与设计方法。

模块独立性是创建良好设计的一个重要原则，一般采用模块间的耦合和模块的内聚两个准则来进行度量。耦合是模块之间的相对独立性的度量，模块之间的连接越紧密，联系越多，耦合性就越高，而其模块独立性就越弱。一般来说，模块之间的耦合有 7 种类型，根据耦合性从低到高为非直接耦合、数据耦合、标记耦合、控制耦合、外部耦合、公共耦合和内容耦合。如果一个模块访问另一个模块时，彼此之间是通过数据参数（不是控制参数、公共数据结构或外部变量）来交换输入、输出信息的，则称这种耦合为数据耦合；如果一组模块通过数据结构本身传递，则称这种耦合为标记耦合；若一组模块都访问同一个公共数据环境，则它们之间的耦合就称为公共耦合；若一个模块直接访问另一个模块的内部数据、一个模块不通过正常入口转到另一个模块内部、两个模块有一部分程序代码重叠或者一个模块有多个入口，上述几个情形之一发生则说明两个模块之间就发生了内容耦合。

参考答案

（16）D

试题（17）

下列关于风险的叙述不正确的是：风险是指 （17） 。

（17）A. 可能发生的事件 B. 一定会发生的事件

C. 会带来损失的事件 D. 可能对其进行干预，以减少损失的事件

试题（17）分析

本题考查风险分析和风险控制技术。

风险是一种具有负面后果的、人们不希望发生的事件。通常认为风险具有以下特点：风险是可能发生的事件，其发生的可能性用风险概率来描述；风险是会给项目带来损失的事件；可能对风险进行干预，以期减少损失。针对每一种风险，应弄清可能减少造成损失或避免损失的程度。对风险加以控制，采取一些有效的措施来降低风险或是消除风险。

参考答案

（17）B

试题（18）

下列关于项目估算方法的叙述不正确的是 （18） 。

（18）A. 专家判断方法受到专家经验和主观性影响

B. 启发式方法（如 COCOMO 模型）的参数难以确定

C．机器学习方法难以描述训练数据的特征和确定其相似性

D．结合上述三种方法可以得到精确的估算结果

试题（18）分析

本题考查项目管理及工具技术。

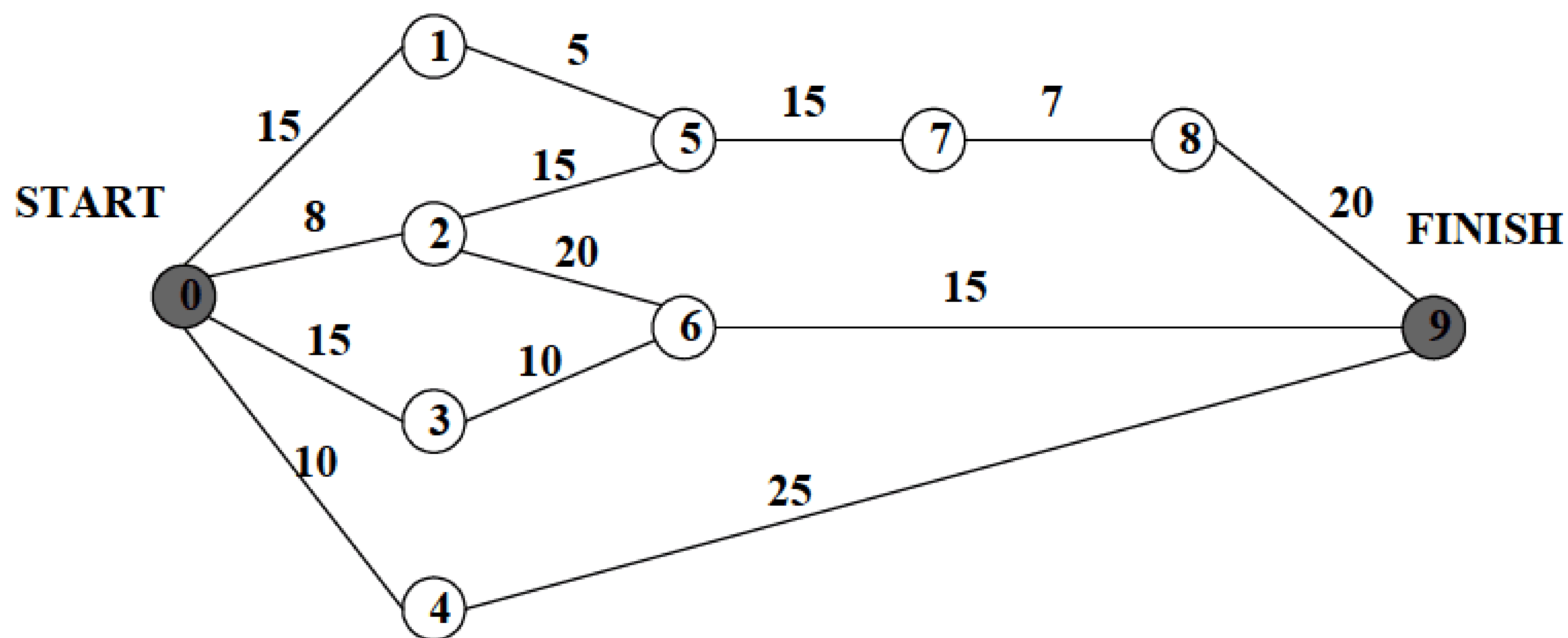
项目估算是项目计划和管理的一个至关重要的方面。成本超出某个限度可能导致客户取消项目，而过低的成本估算可能会迫使开发小组投入大量的时间却没有相应的经济回报。目前常用的项目估算方法有专家判断方法，该方法受到专家经验和主观性等方面的影响；算法方法，根据某个计算模型来估算项目开发成本，如启发式方法 COCOMO 模型，但这些模型中的参数难以确定；机器学习方法，如根据过去的项目开发数据，建立分类模型，预测新项目的开发成本，但这类方法难以定义训练数据的特征以及定义数据对象之间的相似性。即使结合多种方法，上述问题仍然存在，因此并不能得到精确的估算结果。

参考答案

（18）D

试题（19）

下图是一个软件项目的活动图，其中顶点表示项目里程碑，边表示包含的活动，边上的权重表示活动的持续时间，则里程碑 __（19）__ 在关键路径上。



（19）A．1　　B．2　　C．3　　D．4

试题（19）分析

本题考查项目管理及工具技术。

根据关键路径法，计算出关键路径为 0—2—5—7—8—9，关键路径长度为 65。因此里程碑 2 在关键路径上，而里程碑 1、3 和 4 不在关键路径上。

参考答案

（19）B

试题（20）、（21）

算术表达式采用逆波兰式表示时不用括号，可以利用 __（20）__ 进行求值。与逆波兰

式 ab-cd+*对应的中缀表达式是__(21)__。

(20) A. 数组　　B. 栈　　C. 队列　　D. 散列表

(21) A. a-b+c*d　　B. (a-b)* c+d　　C. (a-b)* (c+d)　　D. a-b*c+d

试题（20）、（21）分析

本题考查程序语言的基础知识。

逆波兰式（reverse polish notation，也叫后缀表达式）是将运算符写在操作数之后的表达式表示方法。对逆波兰式进行求值的方法是：从左至右扫描表达式，遇到操作数则压栈，遇到运算符号则从栈中弹出操作数进行运算，然后将运算结果压入栈中，重复该过程直到表达式结束，最后的结果为栈顶元素。由于控制上比较简单，所以逆波兰式更便于计算。

表达式“a-b+c*d”的后缀式为“ab-cd*+”。

表达式“(a-b)* c+d”的后缀式为“ab-c*d+”。

表达式“(a-b)* (c+d)”的后缀式为“ab-cd+*”。

表达式“a-b*c+d”的后缀式为“abc*-d+”。

参考答案

(20) B　　(21) C

试题（22）

若一种程序设计语言规定其程序中的数据必须具有类型，则有利于__(22)__。

① 在翻译程序的过程中为数据合理分配存储单元

② 对参与表达式计算的数据对象进行检查

③ 定义和应用动态数据结构

④ 规定数据对象的取值范围及能够进行的运算

⑤ 对数据进行强制类型转换

(22) A. ①②③　　B. ①②④　　C. ②④⑤　　D. ③④⑤

试题（22）分析

本题考查程序语言的基础知识。

程序中的数据具有类型属性时，就可以规定数据对象的取值范围及能够进行的运算，在运算前便于进行类型检查，也更有利于为数据合理分配存储单元。

参考答案

(22) B

试题（23）、（24）

某文件管理系统在磁盘上建立了位示图（bitmap），记录磁盘的使用情况。若系统的字长为 32 位，磁盘上的物理块依次编号为 0、1、2、…，那么 4096 号物理块的使用情况在位示图中的第__(23)__个字中描述；若磁盘的容量为 200GB，物理块的大小为 1MB，那么位示图的大小为__(24)__个字。

（23）A．129　　B．257　　C．513　　D．1025

（24）A．600　　B．1200　　C．3200　　D．6400

试题（23）、（24）分析

本题考查操作系统文件管理方面的基础知识。

根据题意，系统中字长为 32 位，可记录 32 个物理块的使用情况，这样 0～31 号物理块的使用情况在位示图中的第 1 个字中描述，32～63 号物理块的使用情况在位示图中的第 2 个字中描述，……，4064～4095 号物理块的使用情况在位示图中的第 128 个字中描述，4096～4127 号物理块的使用情况在位示图中的第 129 个字中描述。

根据题意，若磁盘的容量为 200GB，物理块的大小为 1MB，那么该磁盘就有 204 800 个物理块（即 200×1024），位示图的大小为 204 800/32=6400 个字。

参考答案

（23）A　　（24）D

试题（25）、（26）

系统中有 R 类资源 *m* 个，现有 *n* 个进程互斥使用。若每个进程对 R 资源的最大需求为 *w*，那么当 *m*、*n*、*w* 分别取下表中的值时，对于表中的①～⑥种情况，__（25）__可能会发生死锁。若将这些情况的 *m* 分别加上__（26）__，则系统不会发生死锁。

	①	②	③	④	⑤	⑥
m	3	3	5	5	6	6
n	2	3	2	3	3	4
w	2	2	3	3	3	2

（25）A．①②⑤　　B．③④⑤　　C．②④⑤　　D．②④⑥

（26）A．1、1 和 1　　B．1、1 和 2　　C．1、1 和 3　　D．1、2 和 1

试题（25）、（26）分析

本题考查操作系统进程管理方面的基础知识。

试题（25）的正确答案是 C。分析如下：

情况①不会发生死锁：已知系统资源 R 的数目等于 3，进程数等于 2，每个进程对 R 资源的最大需求为 2。若系统为 2 个进程各分配 1 个资源，系统可供分配的剩余资源数等于 1，则可以保证 1 个进程得到所需资源运行完毕。当该进程释放资源后又能保证另一个进程运行完毕，故系统不会发生死锁。

情况②会发生死锁：已知系统资源 R 的数目等于 3，进程数等于 3，每个进程对 R 资源的最大需求为 2。若系统为 3 个进程各分配 1 个资源，系统可供分配的剩余资源数等于 0，则无法保证进程得到所需资源运行完毕，故系统会发生死锁。

情况③不会发生死锁：已知系统资源 R 的数目等于 5，进程数等于 2，每个进程对 R 资源的最大需求为 3。若系统为两个进程各分配两个资源，系统可供分配的剩余资源数

等于 1，则可以保证 1 个进程得到所需资源运行完毕。当该进程释放资源后又能保证另一个进程运行完毕，故系统不会发生死锁。

情况④会发生死锁：已知系统资源 R 的数目等于 5，进程数等于 3，每个进程对 R 资源的最大需求为 3。若系统为 3 个进程分别分配 2、2 和 1 个资源，系统可供分配的剩余资源数等于 0，则无法保证进程得到所需资源运行完毕，故系统会发生死锁。

情况⑤会发生死锁：已知系统资源 R 的数目等于 6，进程数等于 3，每个进程对 R 资源的最大需求为 3。若系统为 3 个进程各分配 2 个资源，系统可供分配的剩余资源数等于 0，则无法保证进程得到所需资源运行完毕，故系统会发生死锁。

情况⑥不会发生死锁：已知系统资源 R 的数目等于 6，进程数等于 4，每个进程对 R 资源的最大需求为 2。若系统为 4 个进程各分配 1 个资源，系统可供分配的剩余资源数等于 2，则可以保证 2 个进程得到所需资源运行完毕。当该进程释放资源后又能保证剩余 2 个进程运行完毕，故系统不会发生死锁。

试题（26）的正确答案是 D。分析如下：

情况②：若将 *m* 加 1 等于 4 时，系统为 3 个进程各分配 1 个资源，系统可供分配的剩余资源数等于 1，则可以保证 1 个进程得到所需资源运行完毕。当该进程释放资源后又能保证另一个进程运行完毕，故系统不会发生死锁。

情况④：若将 *m* 加 2 等于 7 时，系统为 3 个进程各分配 2 个资源，系统可供分配的剩余资源数等于 1，则可以保证 1 个进程得到所需资源运行完毕。当该进程释放资源后又能保证另两个进程运行完毕，故系统不会发生死锁。

情况⑤：若将 *m* 加 1 等于 7 时，系统为 3 个进程各分配 2 个资源，系统可供分配的剩余资源数等于 1，则可以保证 1 个进程得到所需资源运行完毕。当该进程释放资源后又能保证另两个进程运行完毕，故系统不会发生死锁。

参考答案

（25）C　（26）D

试题（27）、（28）

某系统采用请求页式存储管理方案，假设某进程有 6 个页面，系统给该进程分配了 4 个存储块，其页面变换表如下表所示，表中的状态位等于 1/0 分别表示页面在内存/不在内存。当该进程访问的页面 2 不在内存时，应该淘汰表中页号为<u>（27）</u>的页面。假定页面大小为 4K，逻辑地址为十六进制 3C18H，该地址经过变换后的页帧号为<u>（28）</u>。

页　号	页　帧　号	状　态　位	访　问　位	修　改　位
0	5	1	1	1
1	—	0	0	0
2	—	0	0	0
3	2	1	1	0
4	8	1	1	1
5	12	1	0	0

（27）A．0　　B．3　　C．4　　D．5

（28）A．2　　B．5　　C．8　　D．12

试题（27）、（28）分析

本题考查操作系统存储管理方面的基础知识。

在请求页式存储管理方案中，当访问的页面不在内存时需要置换页面，置换页面的原则如下表，即最先置换访问位和修改位为 00 的页，其次是访问位和修改位为 01 的页，然后是访问位和修改位为 10 的页，最后才置换访问位和修改位为 11 的页。因此本题当该进程访问的页面 2 不在内存时，应该淘汰表中页号为 5 的页面。

置换顺序	访问位	修改位
1	0	0
2	0	1
3	1	0
4	1	1

由于 3C18H = 3000 + 0C18，因此该地址对应的页号为 3，根据页面变换表，经变换后的页帧号为 2。

参考答案

（27）D　　（28）A

试题（29）

为了有效地捕获系统需求，应采用__（29）__。

（29）A．瀑布模型　　B．V 模型　　C．原型模型　　D．螺旋模型

试题（29）分析

本题考查软件过程模型。

软件过程是软件生命周期中的一系列相关活动，即用于开发和维护软件及相关产品的一系列活动。软件过程模型可以帮助开发团队理解开发过程，形成对开发中的活动、资源和约束的共同理解，可以根据具体情况对一个过程进行裁剪等。瀑布模型从一种非常高层的角度描述了软件开发过程中进行的活动，并且提出了要求开发人员经过的事件序列。该模型适用于项目开始时需求已确定的情况。V 模型是瀑布模型的变种，它说明测试活动是如何与分析和设计相联系的。原型模型允许开发人员快速地构造整个系统或系统的一部分以理解或澄清问题。原型的用途是获知用户的真正需求，因此原型模型可以有效地引发系统需求。螺旋模型把开发活动和风险管理结合起来，以将风险减到最小并控制风险。

参考答案

（29）C

试题（30）

关于过程改进，以下叙述中不正确的是__（30）__。

（30）A．软件质量依赖于软件开发过程的质量，其中个人因素占主导作用

B．要使过程改进有效，需要制定过程改进目标

C．要使过程改进有效，需要进行培训

D．CMMI 成熟度模型是一种过程改进模型，仅支持阶段性过程改进而不支持连续性过程改进

试题（30）分析

本题考查软件过程改进。

软件开发过程极大地影响所生成的产品质量，因此改进过程将改进软件产品的质量。这也是进行过程改进的前提和理念。软件质量依赖于软件开发过程的质量，其中，人的因素是主导的，开发技术、过程质量、成本时间和进度也是影响因素。另外，要使得过程改进有效，需要制定过程改进的目标，还需要对开发人员进行培训。CMMI 是 SEI 将已有的几个 CMM 模型结合在一起，使之构成“集成模型”，即成熟度模型，该模型支持阶段性过程改进和连续性过程改进。

参考答案

（30）D

试题（31）

软件产品的可靠性并不取决于__（31）__。

（31）A．潜在错误的数量　　B．潜在错误的位置

C．软件产品的使用方式　　D．软件产品的开发方式

试题（31）分析

本题考查软件质量管理。

软件可靠性指的是一个系统对于给定的时间间隔内、在给定条件下无失效运作的概率。根据定义，软件可靠性与软件的潜在错误的数量、位置有关，与软件产品的使用方式有关，而软件产品的开发方式不决定软件产品的可靠性。

参考答案

（31）D

试题（32）

软件__（32）__是指一个系统在给定时间间隔内和给定条件下无失效运行的概率。

（32）A．可靠性　　B．可用性　　C．可维护性　　D．可伸缩性

试题（32）分析

本题考查软件质量管理。

软件可靠性指的是一个系统对于给定的时间间隔内、在给定条件下无失效运作的概率。软件可用性使之在给定的时间点上，一个软件系统能够按照规格说明正确运行的概

率。软件可维护性是在给定的使用条件下，在规定的时间间隔内，使用规定的过程和资源完成维护活动的概率。

参考答案

（32）A

试题（33）

高质量的文档所应具有的特性中，不包括__（33）__。

（33）A．针对性，文档编制应考虑读者对象群

B．精确性，文档的行文应该十分确切，不能出现多义性的描述

C．完整性，任何文档都应当是完整的、独立的，应该自成体系

D．无重复性，同一软件系统的几个文档之间应该没有相同的内容，若确实存在相同内容，则可以用“见**文档**节”的方式引用

试题（33）分析

本题考查文档与软件维护。

文档是指某种数据媒体和其中所记录的数据。在软件开发过程中，有大量的信息要记录和使用，因此文档具有重要的作用，如可以提高软件开发过程的能见度、提高开发效率、作为开发人员在一定阶段的工作成果和结束标志、记录开发过程中的有关信息、提高对软件运行维护和培训的有关信息、便于用户了解软件功能和性能等各项指标。

高质量的文档应该体现在几个方面：针对性，文档编制应考虑读者。按不同的类型、不同层次的读者，决定怎样适应他们的需要；精确性，文档的行文应该十分确切，不能出现多义性的描述。同一项目几个文档的内容应该是协调一致，没有矛盾的；清晰性，文档编写应力求简明，如有可能，配以适当的图表，以增强其清晰性；完整性，任何文档都应当是完整的、独立的，应该自成体系；灵活性，各个不同软件项目，其规模和复杂程度有着许多实际差别，不能一律看待；可追溯性，由于各开发阶段编制的文档与各个阶段完成的工作有密切的关系，前后两个阶段生成的文档，随着开发工作的逐步延伸，具有一定的继承关系，在一个项目各开发阶段之间提供的文档必定存在着可追溯的关系。

参考答案

（33）D

试题（34）

在软件维护阶段，为软件的运行增加监控设施属于__（34）__维护。

（34）A．改正性　　B．适应性　　C．完善性　　D．预防性

试题（34）分析

本题考查软件维护技术。

在软件开发完成交付用户使用后，就进入软件运行/维护阶段。软件维护活动根据其内容可以分为 4 种类型：改正性维护，为了识别和纠正软件错误、改正软件性能上的缺陷、排除实施中的误使用，应进行的诊断和改正错误的过程；适应性维护，由于信息技

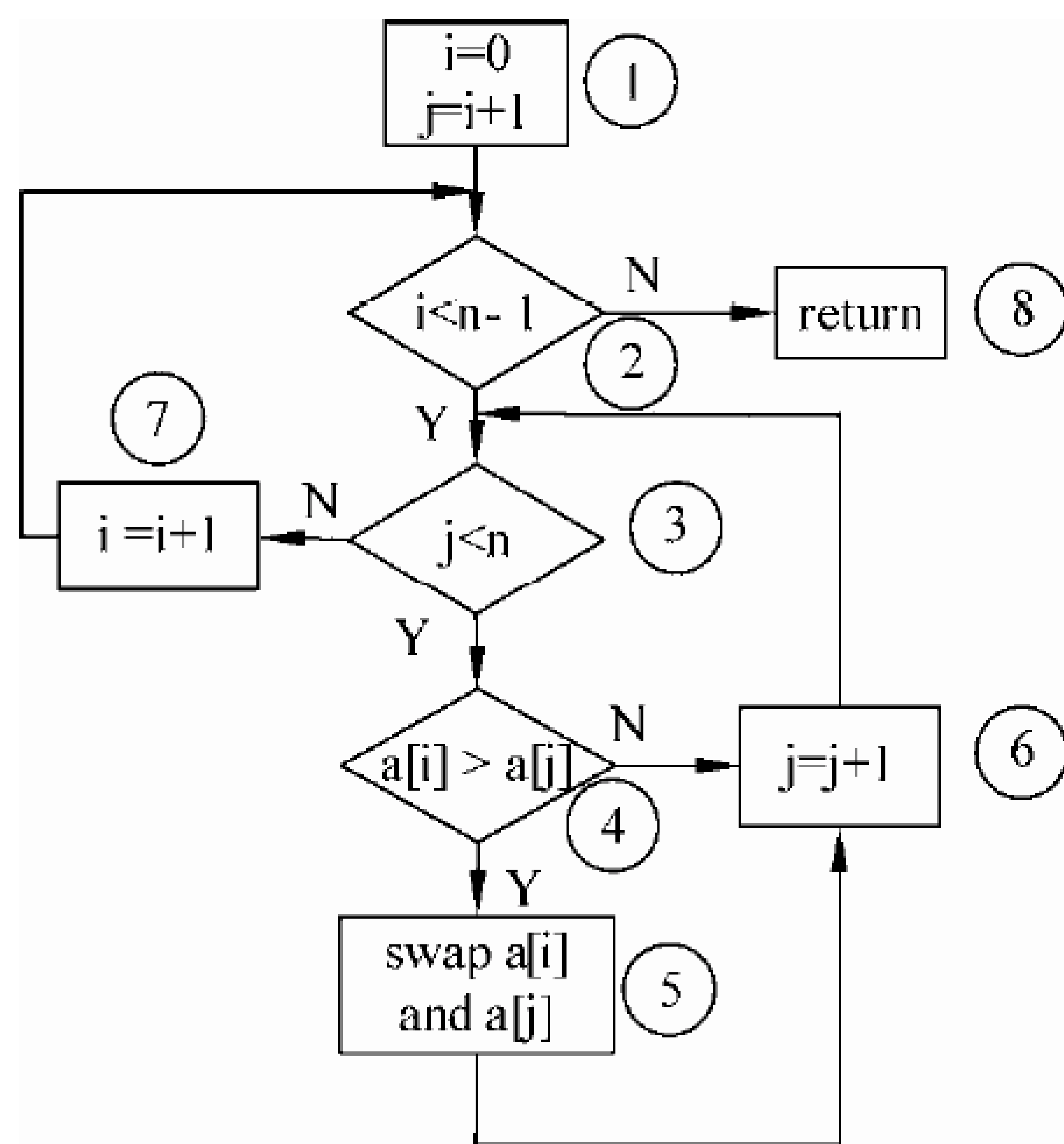
术飞速发展，软件运行的外部环境或数据环境可能会发生变化，为了使软件适应这种变化，而修改软件的过程；完善性维护，在软件使用过程中，用户往往会对软件提出新的功能与性能要求，为了满足这些要求，需要修改或再开发软件，以扩充软件功能、增强软件性能、改进加工效率、提高软件的可维护性而进行的维护活动；预防性维护是为了提高软件的可维护性和可靠性等，为以后进一步改进软件打下良好基础而进行的维护工作。

参考答案

（34）C

试题（35）

下图所示的逻辑流，最少需要__（35）__个测试用例可实现语句覆盖。



（35）A．1　　B．2　　C．3　　D．5

试题（35）分析

本题考查软件测试技术。

语句覆盖是一种白盒测试技术，指的是设计若干测试用例，运行被测程序，使得每一个可执行语句至少执行一次。题中的逻辑流的输入是一个数组 a，只要存在某个 a[i]>a[j] 的情况，则该测试用例下可以覆盖所有的可执行语句，因此至少需要 1 个测试用例即可。

参考答案

（35）A

试题（36）

在改正当前故障的同时可能会引入新的故障，这时需要进行__（36）__。

（36）A．功能测试　　B．性能测试　　C．回归测试　　D．验收测试

试题（36）分析

本题考查软件测试技术。

功能测试检查软件是否能实现需求中指定的那些功能。性能测试是测试软件的安全

性、精确性、速度和可靠性。回归测试用于识别在改正当前故障的同时可能会引入新的故障。验收测试是客户对系统进行测试以验证软件系统是否符合他们对需求的理解。

参考答案

（36）C

试题（37）

面向对象分析的第一步是__（37）__。

（37）A．定义服务　　B．确定附加的系统约束
　　C．确定问题域　　D．定义类和对象

试题（37）分析

本题考查面向对象分析的基本知识。

面向对象分析的目的是为了获得对应用问题的理解，确定系统的功能、性能要求。面向对象分析包含 5 个活动：认定对象、组织对象、描述对象间的相互作用、定义对象的操作和定义对象的内部信息。而分析阶段最重要的是理解问题域的概念，其结果将影响整个工作。经验表明，从应用定义域概念标识对象是非常合理的。因此，面向对象分析的第一步就是确定问题域。

参考答案

（37）C

试题（38）

下列关于一个类的静态成员的描述中，不正确的是__（38）__。

（38）A．类的静态方法只能访问该类的静态数据成员
　　B．静态数据成员可被该类的所有方法访问
　　C．该类的对象共享其静态数据成员的值
　　D．该类的静态数据成员的值不可修改

试题（38）分析

本题考查面向对象开发中静态成员的基本知识。

面向对象开发方法中，静态成员的含义是所修饰的成员是属于类的，而不是属于某对象的。静态数据成员对该类只有一份，该类的所有对象共享静态数据成员，可被该类的所有方法访问，其值可以修改，但是不论是通过对象还是类对静态数据成员值的修改，都会反应到整个类。类的静态方法只能访问该类的静态数据成员。

参考答案

（38）D

试题（39）～（41）

UML 的设计视图包含了类、接口和协作，其中，设计视图的静态方面由__（39）__和__（40）__表现；动态方面由交互图、__（41）__表现。

（39）A．类图　　B．状态图　　C．活动图　　D．序列图

（40）A．交互图 B．对象图 C．通信图 D．定时图

（41）A．状态图和类图 B．类图和活动图

C．对象图和状态图 D．状态图和活动图

试题（39）～（41）分析

本题考查统一建模语言（UML）的基本知识。

通常是用一组视图反映系统的各个方面，以完整地描述系统，每个视图代表系统描述中的一个抽象，显示系统中的一个特定的方面。UML2.0 中提供了多种图形，从静态和动态两个方面表现系统视图。

类图展现了一组对象、接口、协作和它们之间的关系。对象图展现了一组对象以及其之间的关系，描述了在类图中所建立的事物的实例的静态快照。序列图是场景的图形化表示，描述了以时间顺序组织的对象之间的交互活动。通信图和序列图同构，强调收发消息的对象的结构组织。状态图展现了一个状态机，由状态、转换、事件和活动组成，它关注系统的动态视图，强调对象行为的事件顺序。活动图是一种特殊的状态图，展现了在系统内从一个活动到另一个活动的流程，它专注于系统的动态视图。序列图、通信图、交互图和定时图均被称为交互图，它们用于对系统的动态方面进行建模。

参考答案

（39）A （40）B （41）D

试题（42）

UML 中关联的多重度是指__（42）__。

（42）A．一个类中被另一个类调用的方法个数

B．一个类的某个方法被另一个类调用的次数

C．一个类的实例能够与另一个类的多少个实例相关联

D．两个类所具有的相同的方法和属性

试题（42）分析

本题考查面向对象开发的基本知识。

进行面向对象设计时，类图中可以展现类之间的关联关系，还可以在类图中图示关联中的数量关系，即多重度。表示数量关系时，用多重度说明数量或数量范围，表示有多少个实例（对象）能被连接起来，即一个类的实例能够与另一个类的多少个实例相关联。

参考答案

（42）C

试题（43）

在面向对象软件开发过程中，采用设计模式__（43）__。

（43）A．以复用成功的设计

B．以保证程序的运行速度达到最优值

C. 以减少设计过程创建的类的个数

D. 允许在非面向对象程序设计语言中使用面向对象的概念

试题（43）分析

本题考查设计模式的基本知识。

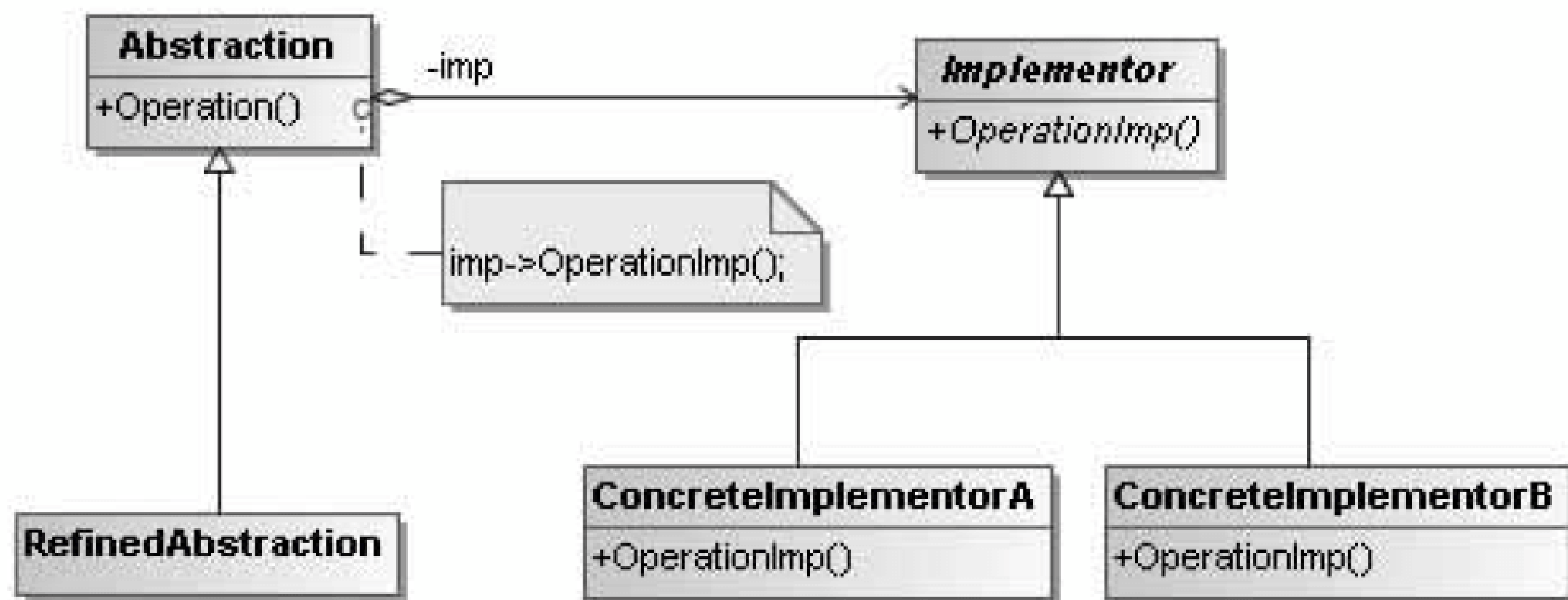
每一个设计模式描述了一个在我们周围不断重复发生的问题，以及该问题的解决方案的核心。这样，就能重复地使用该方案而不必做重复劳动。设计模式的核心在于提供了相关问题的解决方案。因此，面向对象软件开发过程中，采用设计模式的主要目的就是复用成功的设计。

参考答案

（43）A

试题（44）、（45）

设计模式＿（44）＿将抽象部分与其实现部分相分离，使它们都可以独立地变化。下图为该设计模式的类图，其中，＿（45）＿用于定义实现部分的接口。



（44）A. Bridge（桥接）　　B. Composite（组合）

C. Facade（外观）　　D. Singleton（单例）

（45）A. Abstraction　　B. ConcreteImplementorA

C. ConcreteImplementorB　　D. Implementor

试题（44）、（45）分析

本题考查设计模式的基本知识。

Bridge（桥接）模式将对象的抽象和其实现分离，从而可以独立地改变它们，抽象类定义对该抽象的接口，如上图中 Implementor，而具体的子类则用不同方式加以实现，如 ConcreteImplementorA 和 ConcreteImplementorB。Composite（组合）模式是结构型对象模式的一个实例。它描述了如何构造一个类层次式结构，这一结构由两种类型的对象所对应的类构成，其中的组合对象使得用户可以组合基元对象以及其他的组合对象，从而形成任意复杂的结构。Facade（外观）模式则描述了如何用单个对象表示整个子系统。

Singleton（单例）模式保证一个类只产生唯一的一个实例。

参考答案

（44）A　　（45）D

试题（46）

以下关于 Singleton（单例）模式的描述中，正确的是＿（46）＿。

（46）A．它描述了只有一个方法的类的集合

B．它描述了只有一个属性的类的集合

C．它能够保证一个类的方法只能被一个唯一的类调用

D．它能够保证一个类只产生唯一的一个实例

试题（46）分析

本题考查设计模式的基本知识。

例如，通常用户可以对应用系统进行配置，并将配置信息保存在配置文件中，应用系统启动时首先加载配置文件，而这一配置信息在内存中仅有一份。为了保证这一配置实例只有一份，采用 Singleton（单例）模式，以保证一个类只产生唯一的一个实例。

参考答案

（46）D

试题（47）

＿（47）＿将一个类的接口转换成客户希望的另外一个接口，使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。

（47）A．Adapter（适配器）模式　　B．Command（命令）模式

C．Singleton（单例）模式　　D．Strategy（策略）模式

试题（47）分析

本题考查设计模式的基本知识。

Adapter 模式是将类的接口转换成客户希望的另外一个接口，使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。Command 模式将请求封装在对象中，这样它就可作为参数来传递，也可以被存储在历史列表里，或者以其他方式使用。Singleton（单例）模式保证一个类只产生唯一的一个实例。策略模式（Strategy）定义一系列的算法，把它们一个个封装起来，并使它们可以相互替换，这一模式使得算法可以独立于使用它的客户而变化。

参考答案

（47）A

试题（48）

以下关于高级程序设计语言翻译的叙述中，正确的是＿（48）＿。

（48）A．可以先进行语法分析，再进行词法分析

B．在语法分析阶段可以发现程序中的所有错误

C．语义分析阶段的工作与目标机器的体系结构密切相关

D．目标代码生成阶段的工作与目标机器的体系结构密切相关

试题（48）分析

本题考查程序语言处理的基础知识。

将高级语言程序翻译为机器语言程序的过程中，需要依次进行词法分析、语法分析、语义分析、中间代码生成、代码优化和目标代码生成等阶段，其中，中间代码生成和代码优化可以省略。

程序中的错误分为语法错误和语义错误，语法分析阶段不能发现语义错误。

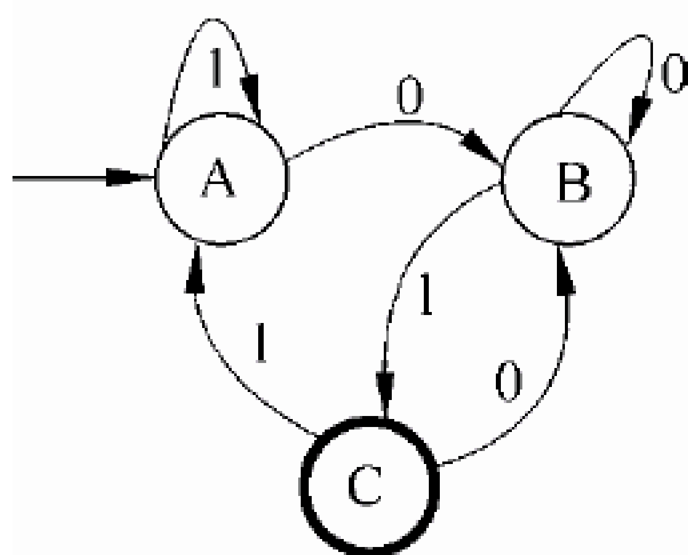
语义分析阶段主要处理语法正确的语言结构的含义信息，可以与目标机器的体系结构无关。目标代码生成阶段的工作与目标机器的体系结构是密切相关的。

参考答案

（48）D

试题（49）

下图所示为一个有限自动机（其中，A 是初态、C 是终态），该自动机可识别＿（49）＿。



（49）A．0000　　B．1111　　C．0101　　D．1010

试题（49）分析

本题考查程序语言处理的基础知识。

从有限自动机的初态到终态的路径上的标记形成其可识别的字符串。

对于题中的自动机，0000 的识别路径为 A→B→B→B→B，不能到达终态 C，所以 0000 不能被该自动机识别；1111 的识别路径为 A→A→A→A→A，不能到达终态 C，所以 1111 也不能被该自动机识别；1010 的识别路径为 A→A→B→C→B，结束状态不是终态 C，所以 1010 不能被该自动机识别；0101 的识别路径为 A→B→C→B→C，存在从初态到终态的识别路径，所以 0101 可以被该自动机识别。

参考答案

（49）C

试题（50）

传值与传地址是函数调用时常采用的信息传递方式，＿（50）＿。

（50）A．在传值方式下，是将形参的值传给实参

B．在传值方式下，形参可以是任意形式的表达式

C．在传地址方式下，是将实参的地址传给形参

D．在传地址方式下，实参可以是任意形式的表达式

试题（50）分析

本题考查程序语言的基础知识。

一个函数被调用时，可能需要接受从外部传入的数据信息，传值调用与引用调用（传地址）是函数调用时常采用的信息传递方式。传值调用是将实参的值传给被调用函数的形参，引用调用的实质是将实参的地址传给被调用函数的形参。

参考答案

（50）C

试题（51）～（54）

某医院数据库的部分关系模式为：科室（科室号，科室名，负责人，电话）、病患（病历号，姓名，住址，联系电话）和职工（职工号，职工姓名，科室号，住址，联系电话）。假设每个科室有一位负责人和一部电话，每个科室有若干名职工，一名职工只属于一个科室；一个医生可以为多个病患看病；一个病患可以由多个医生多次诊治。

科室与职工的所属联系类型为＿（51）＿，病患与医生的就诊联系类型为＿（52）＿。对于就诊联系最合理的设计是＿（53）＿，就诊关系的主键是＿（54）＿。

（51）A．1:1　　B．1:*n*　　C．*n*:1　　D．*n*:*m*

（52）A．1:1　　B．1:*n*　　C．*n*:1　　D．*n*:*m*

（53）A．就诊（病历号，职工号，就诊情况）

B．就诊（病历号，职工姓名，就诊情况）

C．就诊（病历号，职工号，就诊时间，就诊情况）

D．就诊（病历号，职工姓名，就诊时间，就诊情况）

（54）A．病历号，职工号　　B．病历号，职工号，就诊时间

C．病历号，职工姓名　　D．病历号，职工姓名，就诊时间

试题（51）～（54）分析

本题考查数据库基本概念、数据库设计的基础知识。

试题（51）、（52）考查数据库联系类型方面的基本概念。根据题意，“每个科室有若干名职工，一名职工只属于一个科室”，因此科室和职工的所属联系类型是 1:*n*，由“一个医生可以为多个病患看病；一个病患可以由多个医生多次诊治”，得知病患和医生的就诊联系类型是 *n*:*m*。

试题（53）、（54）考查数据库设计方面的基础知识。就诊联系是多对多联系，对于多对多联系只能转换成一个独立的关系模式，关系模式的名称取联系的名称，关系模式的属性取该联系所关联的两个多方实体的码及联系的属性，关系的码是多方实体的码构成的属性组。另外，由于病患会找多个医生为其诊治，因此就诊关系模式设计时需要加上就诊时间，以便唯一区分就诊关系中的每一个元组，即就诊关系模式的主键为（病历

号，职工号，就诊时间）。

参考答案

（51）B　（52）D　（53）C　（54）B

试题（55）、（56）

给定关系模式 $R<U,F>$，$U=\{A,B,C\}$，$F=\{AB\to C, C\to B\}$。关系 R __（55）__，且分别有__（56）__。

（55）A. 只有 1 个候选关键字 AC　B. 只有 1 个候选关键字 AB
C. 有 2 个候选关键字 AC 和 BC　D. 有 2 个候选关键字 AC 和 AB

（56）A. 1 个非主属性和 2 个主属性　B. 2 个非主属性和 1 个主属性
C. 0 个非主属性和 3 个主属性　D. 3 个非主属性和 0 个主属性

试题（55）、（56）分析

本题考查关系数据库规范化理论方面的基础知识。

试题（55）的正确答案是 D。根据函数依赖定义，可知 $AC\to U$，$AB\to U$，所以 AC 和 AB 为候选关键字。

试题（56）的正确答案是 C。根据主属性的定义，“包含在任何一个候选码中的属性叫做主属性（Prime attribute），否则叫做非主属性（Nonprime attribute）”，所以，关系 R 中的 3 个属性都是主属性。

参考答案

（55）D　（56）C

试题（57）

设下三角矩阵（上三角部分的元素值都为 0）$A[0..n,0..n]$如下所示，将该三角矩阵的所有非零元素（即行下标不小于列下标的元素）按行优先压缩存储在容量足够大的数组 $M[\,]$中（下标从 1 开始），则元素 $A[i,j]$（$0\leqslant i\leqslant n, j\leqslant i$）存储在数组 M 的__（57）__中。

$$\begin{bmatrix} A_{0,0} & & & & & \\ A_{1,0} & A_{1,1} & & & 0 & \\ \vdots & & \ddots & & & \\ A_{7,0} & A_{7,1} & A_{7,2} & \cdots & A_{7,7} & \\ A_{8,0} & A_{8,1} & A_{8,2} & A_{8,3} & \cdots & A_{8,8} \end{bmatrix}$$

（57）A. $M\left[\frac{i(i+1)}{2}+j+1\right]$　B. $M\left[\frac{i(i+1)}{2}+j\right]$
C. $M\left[\frac{i(i-1)}{2}+j\right]$　D. $M\left[\frac{i(i-1)}{2}+j+1\right]$

试题（57）分析

本题考查数组存储的基础知识。

按行方式存储时，元素 $A[i,j]$之前的元素个数为$(1+2+\cdots+i+j)$，由于数组 M 的下标从 1 开始，因此，存储 $A[i,j]$的是 $M[1+2+\cdots+i+j+1]$，即 $M[\frac{i(i+1)}{2}+j+1]$。

参考答案

（57）A

试题（58）

对 n 个元素的有序表 $A[1..n]$进行顺序查找，其成功查找的平均查找长度（即在查找表中找到指定关键码的元素时，所进行比较的表中元素个数的期望值）为__（58）__。

（58）A. n　　B. $(n+1)/2$　　C. $\log_2 n$　　D. n^2

试题（58）分析

本题考查顺序查找方法。

假设从前往后找，则所找元素为第 1 个元素时，与表中的 1 个元素作了比较，所找元素为第 2 个元素时，与表中的 2 个元素作了比较，……，所找元素为第 n 个元素时，与表中的 n 个元素作了比较，因此，平均查找长度等于$(1+2+\cdots+n)/n$。

参考答案

（58）B

试题（59）

在__（59）__中，任意一个结点的左、右子树的高度之差的绝对值不超过 1。

（59）A. 完全二叉树　　B. 二叉排序树

C. 线索二叉树　　D. 最优二叉树

试题（59）分析

本题考查二叉树的基本概念。

在平衡二叉树中，任意一个结点的左、右子树的高度之差的绝对值不超过 1。

虽然在结构上都符合二叉树的定义，但完全二叉树、线索二叉树、二叉排序树与最优二叉树的应用场合和概念都不同。

线索二叉树与二叉树的遍历运算相关，是一种存储结构。

二叉排序树的结构与给定的初始关键码序列相关。

最优二叉树（即哈夫曼树）是一类带权路径长度最短的二叉树，由给定的一个权值序列构造。

线索二叉树、二叉排序树和最优二叉树在结构上都不要求是平衡二叉树。

在完全二叉树中，去掉最后一层后就是满二叉树，而且最后一层上的叶子结点必须从该层的最左边开始排列，满足任意一个结点的左、右子树的高度之差的绝对值不超过 1 的条件，因此在形态上是一个平衡的二叉树。

参考答案

（59）A

试题（60）

设一个包含 N 个顶点、E 条边的简单无向图采用邻接矩阵存储结构（矩阵元素 $A[i][j]$ 等于 1/0 分别表示顶点 i 与顶点 j 之间有/无边），则该矩阵中的非零元素数目为＿（60）＿。

（60）A．N　　B．E　　C．$2E$　　D．$N+E$

试题（60）分析

本题考查数据结构的基础知识。

无向图的邻接矩阵是一个对称矩阵，每条边会表示两次，因此矩阵中的非零元素数目为 $2E$。

参考答案

（60）C

试题（61）

对于关键字序列（26,25,72,38,8,18,59），采用散列函数 H(Key)=Key mod 13 构造散列表（哈希表）。若采用线性探测的开放定址法解决冲突（顺序地探查可用存储单元），则关键字 59 所在散列表中的地址为＿（61）＿。

（61）A．6　　B．7　　C．8　　D．9

试题（61）分析

本题考查散列表的基本概念。

对于关键字序列（26,25,72,38,8,18,59）和散列函数 H(Key)=Key mod 13，采用线性探测的开放定址法解决冲突构造的散列表如下表所示：

0	1	2	3	4	5	6	7	8	9	10	11	12
26	38				18		72	8	59			25

参考答案

（61）D

试题（62）

要在 8×8 的棋盘上摆放 8 个“皇后”，要求“皇后”之间不能发生冲突，即任何两个“皇后”不能在同一行、同一列和相同的对角线上，则一般采用＿（62）＿来实现。

（62）A．分治法　　B．动态规划法

C．贪心法　　D．回溯法

试题（62）分析

本题考查算法设计技术。

N-皇后问题是一个经典的计算问题，该问题基于一些约束条件来求问题的可行解。该问题不易划分为子问题求解，因此分治法不适用；由于不是要求最优解，因此不具备最优子结构性质，也不宜用动态规划法和贪心法求解。而系统搜索法——回溯法可以有效地求解该问题。

参考答案

（62）D

试题（63）

分治算法设计技术__（63）__。

（63）A．一般由三个步骤组成：问题划分、递归求解、合并解

B．一定是用递归技术来实现

C．将问题划分为 k 个规模相等的子问题

D．划分代价很小而合并代价很大

试题（63）分析

本题考查算法设计技术。

分治方法是一种重要的算法设计技术（设计策略），该策略将原问题划分成 n 个规模较小而结构与原问题相似的子问题；递归地解决这些子问题；然后再合并其结果，最终得到原问题的解。分治算法往往用递归技术来实现，但并非必须。分治算法最理想的情况是划分为 k 个规模相等的子问题，但很多时候往往不能均匀地划分子问题。分治算法的代价在划分子问题和合并子问题的解上，根据不同的问题，划分的代价和合并的代价有所不同。例如归并排序中，主要的计算代价在合并解上，而在快速排序中，主要的计算代价在划分子问题上。

参考答案

（63）A

试题（64）

某算法的时间复杂度可用递归式 $T(n)=\begin{cases}\Theta(1) & ,n=1\\ 6T(n/5)+n & ,n>1\end{cases}$ 表示，若用 Θ 表示，则正确的是__（64）__。

（64）A．$\Theta(n^{\log_5^6})$　　B．$\Theta(n^2)$　　C．$\Theta(n)$　　D．$\Theta(n^{\log_6^5})$

试题（64）分析

本题考查算法分析技术。

用主定理可以很容易算出该递归式。主定理给出了求解形如 $T(n)=aT(n/b)+f(n)$ 的递归式的一般方法。比较 $n^{\log_b^a}$ 和 $f(n)$ 中 n 的最高次幂的关系，考虑三种情况：若存在某常数 $\varepsilon>0$，有 $f(n)=O(n^{\log_b^a-\varepsilon})$，则 $T(n)=\Theta(n^{\log_b^a})$；若 $f(n)=O(n^{\log_b^a}\lg^k n)$，则 $T(n)=\Theta(n^{\log_b^a}\lg^{k+1} n)$；若 $f(n)=O(n^{\log_b^a+\varepsilon})$，则 $T(n)=\Theta(f(n))$。题中，a=6，b=5，属于第一种情况，因此有 $T(n)=\Theta(n^{\log_b^a})=\Theta(n^{\log_5^6})$。

参考答案

（64）A

试题（65）

用插入排序和归并排序算法对数组<3,1,4,1,5,9,6,5>进行从小到大排序，则分别需要进行__（65）__次数组元素之间的比较。

（65）A．12,14　　B．10,14　　C．12,16　　D．10,16

试题（65）分析

本题考查排序算法。

插入排序算法的基本思想是将待排序数组分为两个部分，已排好序部分和未排序部分。其主要步骤为：开始时，第一个元素在已排好序部分中，其余元素在未排序部分。然后依次从未排序部分中取出第一个元素，从后向前与排好序部分的元素进行比较并将其插入到已排好序部分的正确位置。直到所有元素排好序。

归并排序的基本思想是将待排序数组划分为子问题，对子问题求解，然后合并解。其主要步骤为：将数组分为两个相同规模的子数组，分别包含前 $n/2$ 个元素和后 $n/2$ 个元素；递归地排序这两个子数组；合并排好序的两个子数组，依次比较两个排好序的子数组的元素，得到整个数组的排好序的序列。

根据上述算法思想和算法步骤，可以得到题中实例的比较次数分别为 12 和 14。

参考答案

（65）A

试题（66）、（67）

ARP 协议属于__（66）__协议，它的作用是__（67）__。

（66）A．物理层　　B．数据链路层　　C．网络层　　D．传输层

（67）A．实现 MAC 地址与主机名之间的映射

B．实现 IP 地址与 MAC 地址之间的变换

C．实现 IP 地址与端口号之间的映射

D．实现应用进程与物理地址之间的变换

试题（66）、（67）分析

ARP 是网络层协议，它的作用是实现 IP 地址与 MAC 地址之间的变换。IP 地址是分配给主机的逻辑地址，在互联网中表示唯一的主机。另外，每个主机还有一个物理地址，通常用网卡地址（MAC 地址）来表示主机的物理地址。

物理地址和逻辑地址的区别可以从两个角度看：从网络互连的角度看，逻辑地址在整个互连网络中有效，而物理地址只是在子网内部有效；从网络协议分层的角度看，逻辑地址由 Internet 层使用，而物理地址由子网访问子层（具体地说就是数据链路层）使用。

由于有两种主机地址，因而需要一种映像关系把这两种地址对应起来。在 Internet 中用地址分解协议（Address Resolution Protocol，ARP）来实现逻辑地址到物理地址映像。ARP 分组的格式如下图所示。

硬件类型		协议类型
硬件地址长度	协议地址长度	操作类型
发送结点硬件地址		
发送结点协议地址		
目标结点硬件地址		
目标结点协议地址		

各字段的含义解释如下：

- 硬件类型：网络接口硬件的类型，对以太网此值为 1。
- 协议类型：发送方使用的协议，0800H 表示 IP 协议。
- 硬件地址长度：对以太网，地址长度为 6 字节。
- 协议地址长度：对 IP 协议，地址长度为 4 字节。
- 操作类型：1—ARP 请求，2—ARP 响应，3—RARP 请求，4—RARP 响应。

通常 Internet 应用程序把要发送的报文交给 IP 协议，IP 当然知道接收方的逻辑地址（否则就不能通信了），但不一定知道接收方的物理地址。在把 IP 分组向下传送给本地数据链路实体之前可以用两种方法得到目标物理地址：

① 查本地内存中的 ARP 地址映像表，其逻辑结构如下表所示。可以看出这是 IP 地址和以太网地址的对照表。

② 如果在 ARP 表中查不到，就广播一个 ARP 请求分组，这种分组经过路由器进一步转发，可以到达所有连网的主机。它的含义是“如果你的 IP 地址是这个分组中的目标结点协议地址，请回答你的物理地址是什么”。收到该分组的主机一方面可以用分组中的两个源地址更新自己的 ARP 地址映像表，另一方面用自己的 IP 地址与目标结点协议地址字段比较，若相符则发回一个 ARP 响应分组，向发送方报告自己的硬件地址，若不相符则不予回答。

IP 地址	以太网地址
130.130.87.1	08 00 39 00 29 D4
129.129.52.3	08 00 5A 21 17 22
192.192.30.5	08 00 10 99 A1 44

参考答案

（66）C　　（67）B

试题（68）

下面关于集线器与交换机的描述中，错误的是__（68）__。

（68）A．交换机是一种多端口网桥

B．交换机的各个端口形成一个广播域

C．集线器的所有端口组成一个冲突域

D．集线器可以起到自动寻址的作用

试题（68）分析

集线器是一种物理层设备，它的作用是从一个端口接收信息，并向其他端口广播出去。集线器不解释所传送信息的含义，也不能识别任何协议数据单元。集线器的各个端口构成一个冲突域，即只能有一个端口发送数据，如果有两个以上端口同时发送，就冲突了。网桥是数据链路层设备，能识别数据链路层协议数据单元，并根据数据链路层地址进行数据转发。交换机是一种多端口网桥，任何一对端口之间都能进行数据转发。交换机的各个端口构成一个广播域，但不是冲突域，即可以有多个端口同时发送数据而不会出现冲突。

参考答案

（68）D

试题（69）

“三网合一”的三网是指__（69）__。

（69）A．电信网、广播电视网、互联网　　B．物联网、广播电视网、电信网

C．物联网、广播电视网、互联网　　D．物联网、电信网、互联网

试题（69）分析

“三网合一”是将电信网、广播电视网以及互联网进行整合，实现业务互联互通的一种网络解决方案。

参考答案

（69）A

试题（70）

要使 4 个连续的 C 类网络汇聚成一个超网，则子网掩码应该为__（70）__。

（70）A．255.240.0.0　　B．255.255.0.0

C．255.255.252.0　　D．255.255.255.252

试题（70）分析

把 4 个 C 类网络汇聚成一个超网地址，使用的网络掩码为 255.255.252.0。

参考答案

（70）C

试题（71）～（75）

Ravi, like many project __（71）__ , had studied the waterfall model of software development as the primary software life-cycle __（72）__ . He was all set to use it for an upcoming project, his first assignment. However, Ravi found that the waterfall model could not be used because the customer wanted the software delivered in stages, something that implied that the system had to be delivered and built in __（73）__ and not as __（74）__ .

The situation in many other projects is not very different. The real world rarely presents a

problem in which a standard process, or the process used in a previous project, is the best choice. To be the most suitable, an existing process must be ＿（75）＿ to the new problem.

A development process, even after tailoring, generally cannot handle change requests. To accommodate change requests without losing control of the project, you must supplement the development process with a requirement change management process.

（71）A. customers　B. managers　C. users　D. administrators
（72）A. activity　B. procedure　C. process　D. progress
（73）A. parts　B. modules　C. software　D. a whole
（74）A. parts　B. modules　C. software　D. a whole
（75）A. modified　B. used　C. suited　D. tailored

试题（71）～（75）分析

本题考查英语的基本知识。

和许多项目经理一样，Ravi 研究了作为主要软件开发生命周期过程的瀑布模型，但是，他发现瀑布模型不能满足要求，原因是客户希望软件分阶段提交。也就说明系统必须按照部分构建和交付系统，而不是作为一个整体进行。这种情况在很多其他项目中也类似。现实世界中，很难有一种标准的过程或在前期的项目中使用的过程作为目前项目的最佳选择。因此，为了达到最佳的适应性，需要针对新的问题，对已有开发过程进行裁剪（针对新的问题，做适应性修改）。但是，即使经过裁剪，一个开发过程也很难应对变更的需求。因此，为了适应变化的需求而不失去对项目的控制，必须用需求变更管理过程对开发过程进行补充。

参考答案

（71）B　（72）C　（73）A　（74）D　（75）D

第 10 章　2011 上半年软件设计师下午试题分析与解答

试题一（共 15 分）

阅读下列说明和图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

【说明】

某医院欲开发病人监控系统。该系统通过各种设备监控病人的生命特征，并在生命特征异常时向医生和护理人员报警。该系统的主要功能如下：

（1）本地监控：定期获取病人的生命特征，如体温、血压、心率等数据。

（2）格式化生命特征：对病人的各项重要生命特征数据进行格式化，然后存入日志文件并检查生命特征。

（3）检查生命特征：将格式化后的生命特征与生命特征范围文件中预设的正常范围进行比较。如果超出了预设范围，系统就发送一条警告信息给医生和护理人员。

（4）维护生命特征范围：医生在必要时（如新的研究结果出现时）添加或更新生命特征值的正常范围。

（5）提取报告：在医生或护理人员请求病人生命特征报告时，从日志文件中获取病人生命特征生成特征报告，并返回给请求者。

（6）生成病历：根据日志文件中的生命特征，医生对病人的病情进行描述，形成病历存入病历文件。

（7）查询病历：根据医生的病历查询请求，查询病历文件，给医生返回病历报告。

（8）生成治疗意见：根据日志文件中的生命特征和病历，医生给出治疗意见，如处方等，并存入治疗意见文件。

（9）查询治疗意见：医生和护理人员查询治疗意见，据此对病人进行治疗。

现采用结构化方法对病人监控系统进行分析与设计，获得如图 1-1 所示的顶层数据流图和图 1-2 所示的 0 层数据流图。

【问题 1】（3 分）

使用说明中的词语，给出图 1-1 中的实体 E1～E3 的名称。

【问题 2】（4 分）

使用说明中的词语，给出图 1-2 中的数据存储 D1～D4 的名称。

【问题 3】（6 分）

图 1-2 中缺失了 4 条数据流，使用说明、图 1-1 和图 1-2 中的术语，给出数据流的名称及其起点和终点。

【问题 4】（2 分）

说明实体 E1 和 E3 之间可否有数据流，并解释其原因。

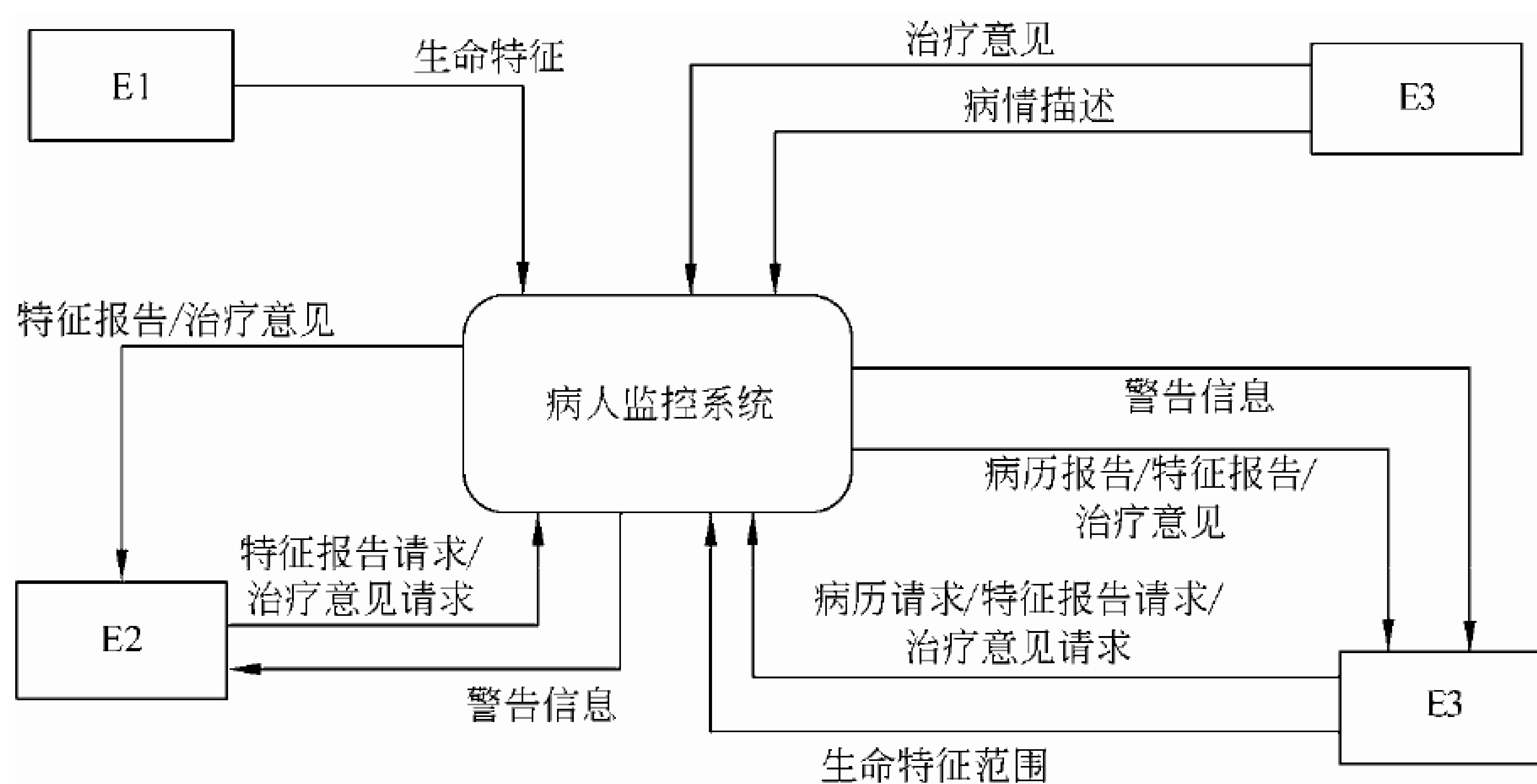


图 1-1　顶层数据流图

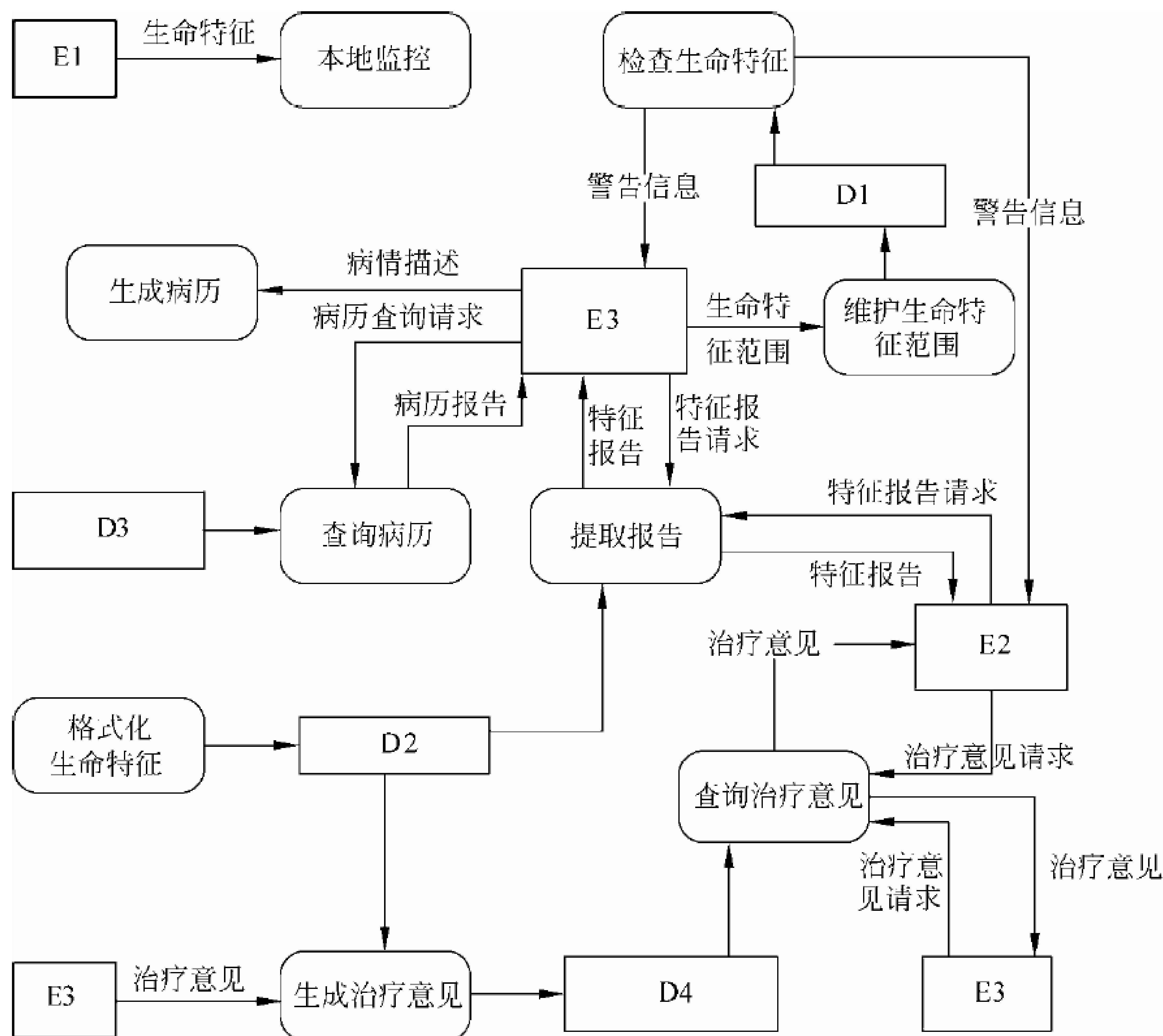


图 1-2　0 层数据流图

试题一分析

本题考查数据流图（DFD）应用于采用结构化方法进行系统分析与设计，是比较传

统的题目，要求考生细心分析题目中所描述的内容。

DFD 是一种便于用户理解、分析系统数据流程的图形化建模工具，是系统逻辑模型的重要组成部分。

【问题 1】

本问题考查顶层 DFD。顶层 DFD 一般用来确定系统边界，将待开发系统看作一个加工，因此图中只有唯一的一个处理和一些外部实体，以及这两者之间的输入输出数据流。题目要求根据描述来确定图中的外部实体。分析题目中的描述，并结合已经在顶层数据流图中给出的数据流进行分析。从中可以看出，与系统的交互者包括病人、医生和护理人员。其中，本地监控定期获取病人的生命特征，病人是生命特征数据来源，医生和护理人员提出相关请求，并得到相关报告结果，如请求病人生命特征报告，并获得相关报告。医生还需要在必要时添加或更新生命特征范围。对应图 1-1 中数据流和实体的对应关系，可知 E1 为病人，E2 为护理人员，E3 为医生。

【问题 2】

本问题考查 0 层 DFD 中数据存储的确定。根据说明中描述：（2）格式化生命特征：对病人的各项重要生命特征数据进行格式化，然后存入日志文件并检查生命特征；（4）维护生命特征范围：医生在必要时（如新的研究结果出现时）添加或更新生命特征值的正常范围；（6）生成病历：根据日志文件中的生命特征，医生对病人的病情进行描述，形成病历存入病历文件；（8）生成治疗意见：根据日志文件中的生命特征和病历，医生给出治疗意见，如处方等，并存入治疗意见文件。因此，D1 为生命特征范围文件，D2 为日志文件，D3 为病例文件，D4 为治疗意见文件。

【问题 3】

本问题考查 0 层 DFD 中缺失的处理和数据流。从说明中的描述及图 1-2 可知，本地监控之后要对重要生命特征存储日志文件并进行格式化，所以在本地监控和格式化生命特征之间缺少了数据流重要生命特征；检查生命特征是对格式化后的生命特征进行检查，所以在格式化生命特征和检查生命特征之间缺少了数据流格式化后的生命特征；根据日志文件中的生命特征，医生对病人的病情进行描述，形成病历存入病历文件。

【问题 4】

本问题考查绘制 DFD 时的注意事项。在 DFD 中，每条数据流的起点和终点之一必须是加工（处理）。本题中，医生和护理人员根据查询到的治疗意见对病人进行治疗属于系统之外的行为，所以两个实体之间不可以有数据流。

参考答案

【问题 1】

E1：病人　　　　E2：护理人员　　　　E3：医生

【问题 2】

D1：生命特征范围文件　　　　D2：日志文件

D3：病历文件　　　　　　　　D4：治疗意见文件

【问题 3】

数据流名称	起　　点	终　　点
重要生命特征	本地监控	格式化生命特征
格式化后的生命特征	格式化生命特征	检查生命特征
病例	生成病历	D3 或 病历（文件）
生命特征	D2 或 日志（文件）	生成病例

【问题 4】

E1 和 E3 之间不可以有数据流，因为数据流的起点和终点中必须有一个是加工（处理）。

试题二（共 15 分）

阅读下列说明，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某服装销售公司拟开发一套服装采购管理系统，以方便对服装采购和库存进行管理。

【需求分析】

（1）采购系统需要维护服装信息及服装在仓库中的存放情况。服装信息主要包括：服装编码、服装描述、服装类型、销售价格、尺码和面料，其中，服装类型为销售分类，服装按销售分类编码。仓库信息包括：仓库编码、仓库位置、仓库容量和库管员。系统记录库管员的库管员编码、姓名和级别。一个库管员可以管理多个仓库，每个仓库有一名库管员。一个仓库中可以存放多类服装，一类服装可能存放在多个仓库中。

（2）当库管员发现有一类或者多类服装缺货时，需要生成采购订单。一个采购订单可以包含多类服装。每类服装可由多个不同的供应商供应，但具有相同的服装编码。采购订单主要记录订单编码、订货日期和应到货日期，并详细记录所采购的每类服装的数量、采购价格和对应的多个供应商。

（3）系统需记录每类服装的各个供应商信息和供应商生产服装的情况。供应商信息包括：供应商编码、供应商名称、地址、企业法人和联系电话。一个供应商可以供应多类服装，一类服装可由多个供应商供应。库管员根据入库时的服装质量情况，设定每个供应商所供应的每类服装的服装质量等级，作为后续采购服装时，选择供应商的参考标准。

【概念模型设计】

根据需求阶段收集的信息，设计的实体联系图（不完整）如图 2-1 所示。

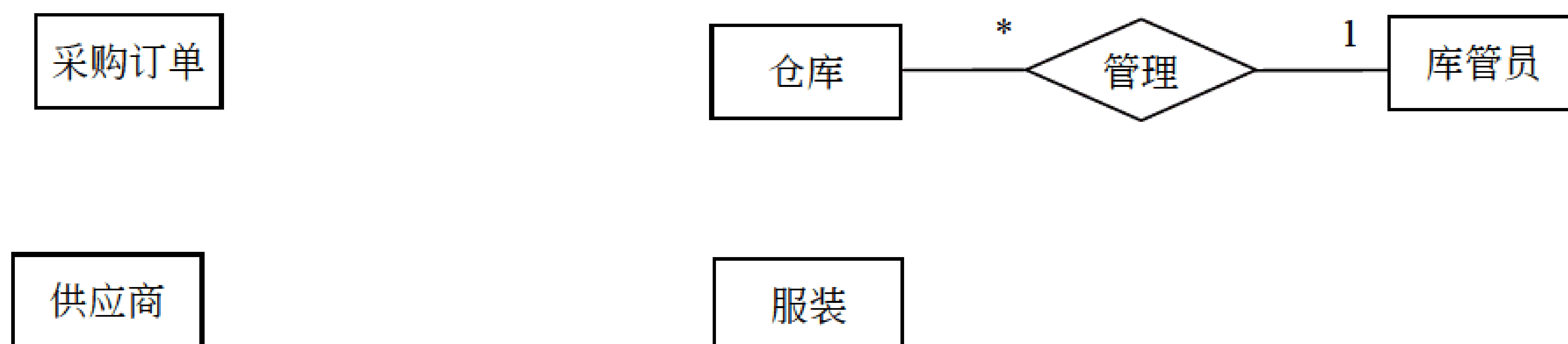


图 2-1　实体联系图

【逻辑结构设计】

根据概念模型设计阶段完成的实体联系图，得出如下关系模式（不完整）：

库管员（库管员编码，姓名，级别）
仓库信息（＿＿（1）＿＿，仓库位置，仓库容量）
服装（服装编码，服装描述，服装类型，尺码，面料，销售价格）
供应商（供应商编码，供应商名称，地址，联系电话，企业法人）
供应情况（＿＿（2）＿＿，服装质量等级）
采购订单（＿＿（3）＿＿）
采购订单明细（＿＿（4）＿＿）

【问题 1】（6 分）

根据需求分析的描述，补充图 2-1 中的联系和联系的类型。

【问题 2】（6 分）

根据补充完整的图 2-1，将逻辑结构设计阶段生成的关系模式中的空（1）～（4）补充完整，并给出其主键（用下划线指出）。

【问题 3】（3 分）

如果库管员定期需要轮流对所有仓库中的服装质量进行抽查，对每个仓库中的每一类被抽查服装需要记录一条检查结果，并且需要记录抽查的时间和负责抽查的库管员。请根据该要求，对图 2-1 进行修改，画出修改后的实体间联系和联系的类型。

试题二分析

本题考查数据库设计，属于比较传统的题目，考查点也与往年类似。

【问题 1】

本问题考查数据库的概念结构设计，题目要求补充完整实体联系图中的联系和联系的类型。

根据题目的需求描述可知，一个库管员可以管理多个仓库，每个仓库有一名库管员。所以，仓库实体和库管员实体之间存在“管理”联系，联系的类型为多对一（*:1）。

根据题目的需求描述可知，一个仓库中可以存放多类服装，一类服装可能存放在多个仓库中。所以，仓库实体和服装实体之间存在“存放”联系，联系的类型为多对多（*:*）。

根据题目的需求描述可知，一个采购订单可以包含多类服装，每类服装可由多个不同的供应商供应。所以，采购订单实体与服装实体和供应商实体三者之间存在“采购”联系，三者之间联系的类型为多对多对多（*:*:*）。

根据题目的需求描述可知，一个供应商可以供应多类服装，一类服装可由多个供应商供应。所以，供应商实体和服装实体之间存在“供应”联系，联系的类型为多对多（*:*）。

【问题 2】

本问题考查数据库的逻辑结构设计，题目要求补充完整各关系模式，并给出各关系模式的主键。

根据实体联系图和需求描述，系统记录库管员的库编码、姓名和级别。所以，对于“库管员”关系模式，需补充属性“库管员编码”。

根据实体联系图和需求描述，仓库信息主要包括：仓库编码、仓库位置、仓库容量和库管员。对于“仓库信息”关系模式，由于仓库实体与库管员实体有多对一联系，需记录对应的库管员，并且需补充属性——仓库编码。因此，“仓库信息”关系模式，需补充属性“仓库编码”和“库管员编码”。

根据实体联系图和需求描述，供应商信息包括：供应商编码、供应商名称、地址、企业法人和联系电话。所以，对于“供应商”关系模式，需补充属性“供应商编码”。

根据实体联系图和需求描述，“供应情况”关系模式，需记录供应商和服装的多对多联系，即一个供应商可以供应多类服装，一类服装可由多个供应商供应。所以，对于“供应商”关系模式，需补充属性“供应商编码”和“服装编码”。

根据实体联系图和需求描述，采购订单主要记录订单编码、订货日期和应到货日期。所以，对于“采购订单”关系模式需补充属性：订单编码、订货日期和应到货日期。由于采购订单还需详细记录所采购的每类服装的数量、采购价格和对应的多个供应商。因此，“采购订单明细”关系模式，需记录采购订单实体与服装实体和供应商实体三者之间存在的多对多对多联系。对于“采购订单明细”关系模式，需补充属性：订单编码、服装编码、供应商编码、数量和采购价格。

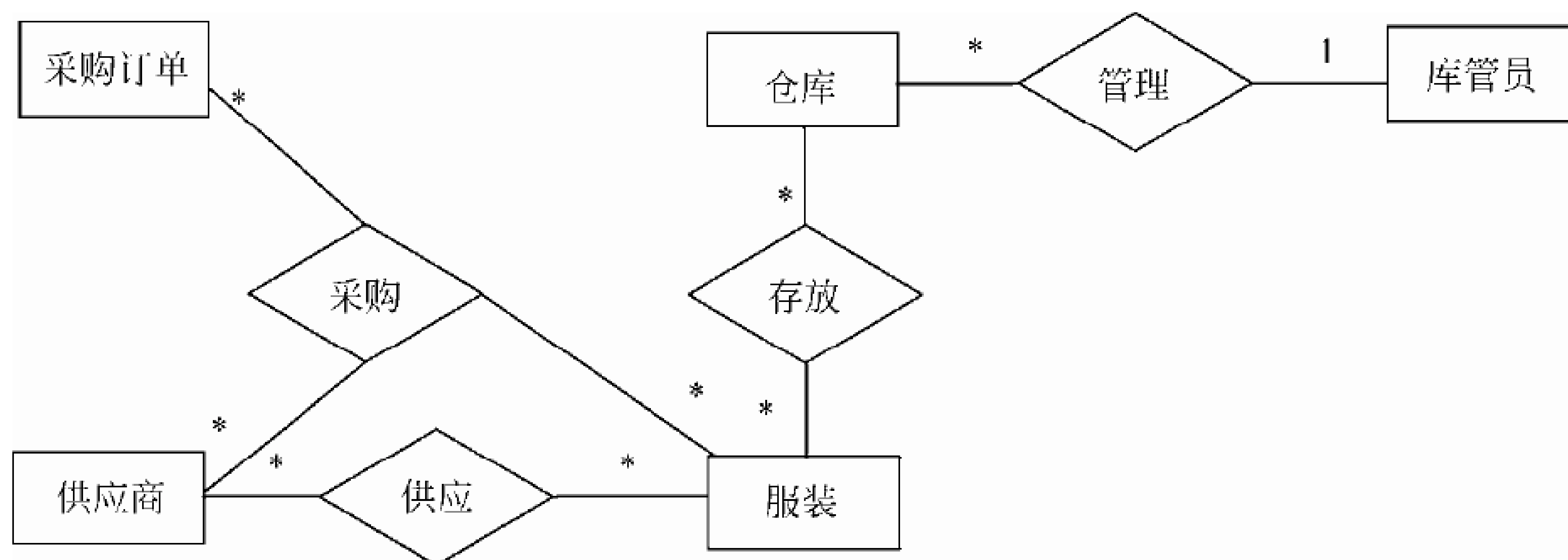
【问题 3】

本问题考查的是数据库的概念结构设计，根据新增的需求增加实体联系图中的实体的联系和联系的类型。

根据问题描述，多个库管员需对每个仓库中的每一类被抽查服装记录一条抽查结果。则在库管员实体与仓库实体和服装实体三者之间存在“抽查”联系，联系的类型是多对多对多(*:*:*)。

参考答案

【问题 1】



【问题 2】

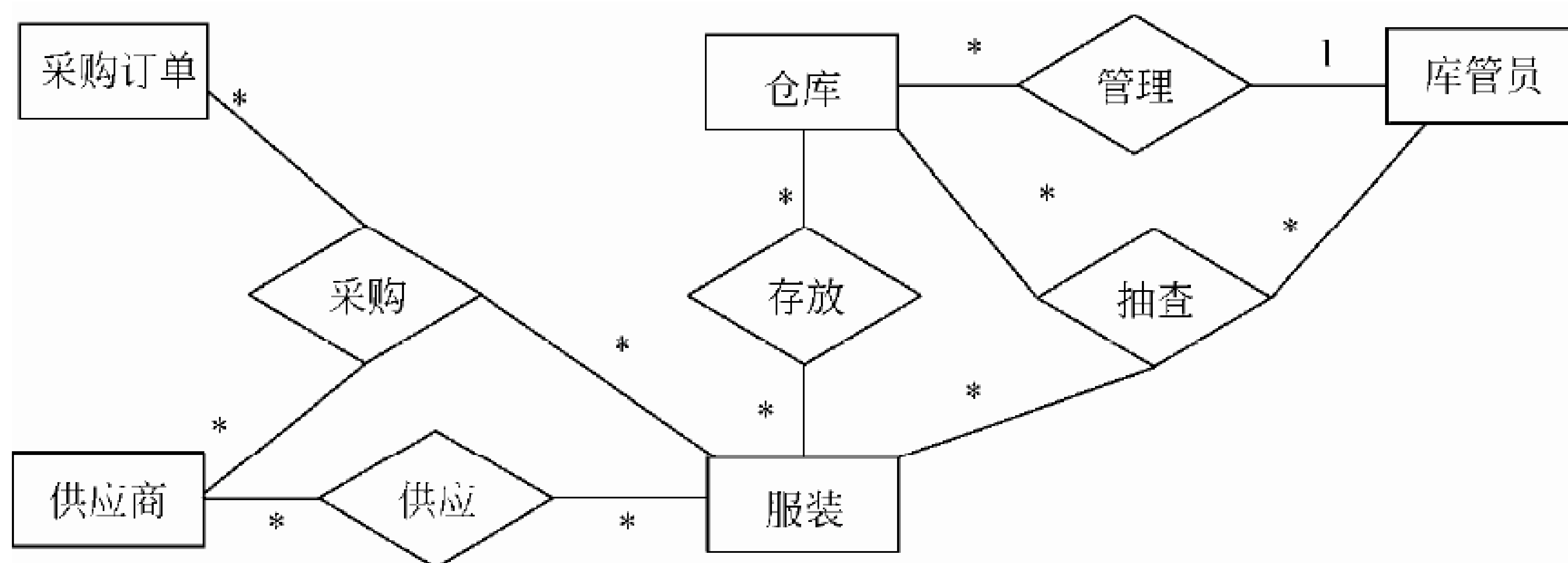
（1）仓库编码，库管员编码

（2）供应商编码，服装编码

（3）订单编码，订货日期，应到货日期

（4）订单编码，服装编码，供应商编码，数量，采购价格

【问题 3】



试题三（共 15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

一个简单的图形编辑器提供给用户的基本操作包括：创建图形、创建元素、选择元素以及删除图形。图形编辑器的组成及其基本功能描述如下：

（1）图形由文本元素和图元元素构成，图元元素包括线条、矩形和椭圆。

（2）图形显示在工作空间中，一次只能显示一张图形（即当前图形，current）。

（3）编辑器提供了两种操作图形的工具：选择工具和创建工具。对图形进行操作时，一次只能使用一种工具（即当前活动工具，active）。

① 创建工具用于创建文本元素和图元元素。

② 对于显示在工作空间中的图形，使用选择工具能够选定其中所包含的元素，可以选择一个元素，也可以同时选择多个元素。被选择的元素称为当前选中元素（selected）。

③ 每种元素都具有对应的控制点。拖曳选定元素的控制点，可以移动元素或者调整元素的大小。

现采用面向对象方法开发该图形编辑器，使用 UML 进行建模。构建出的用例图和类图分别如图 3-1 和图 3-2 所示。

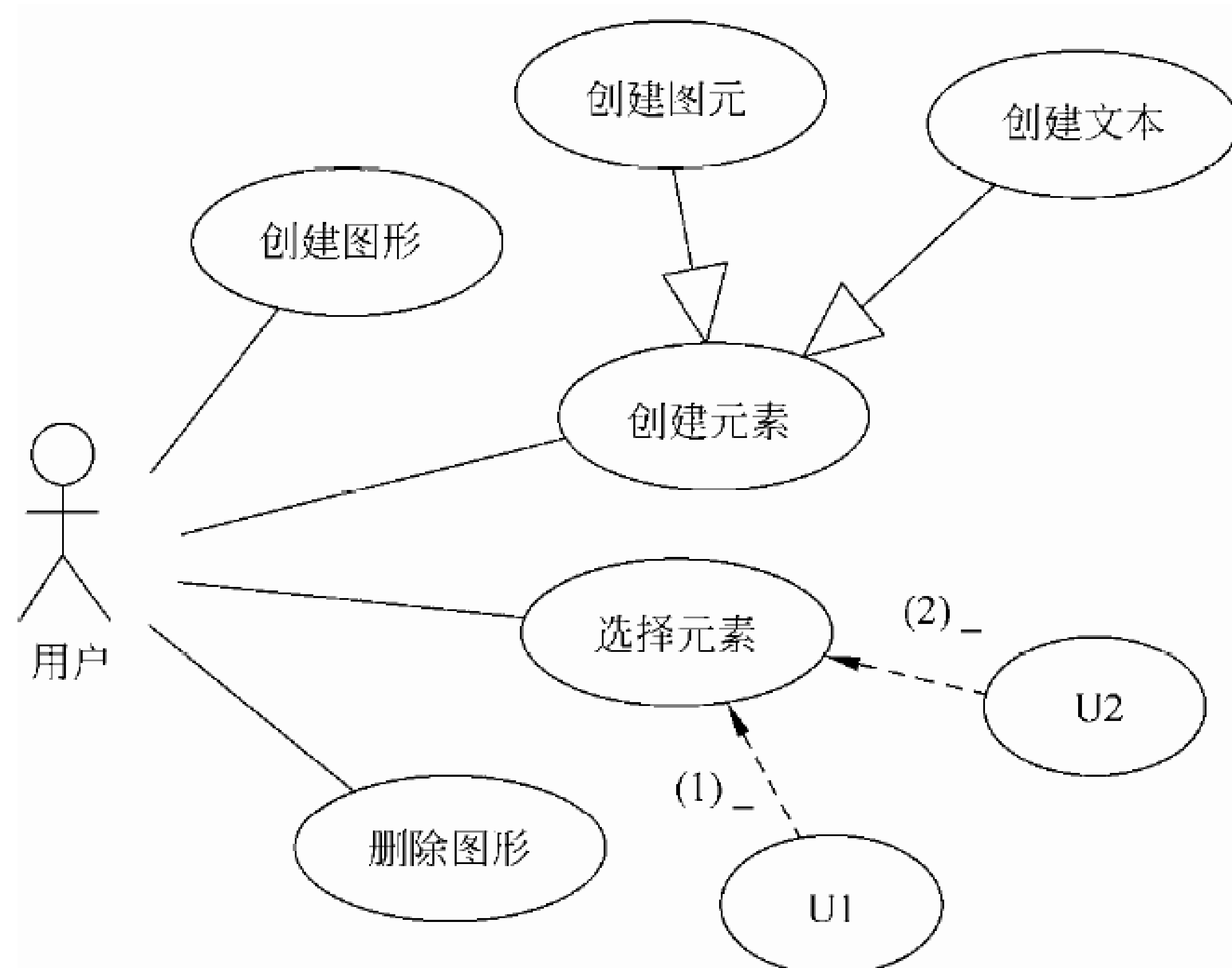


图 3-1 用例图

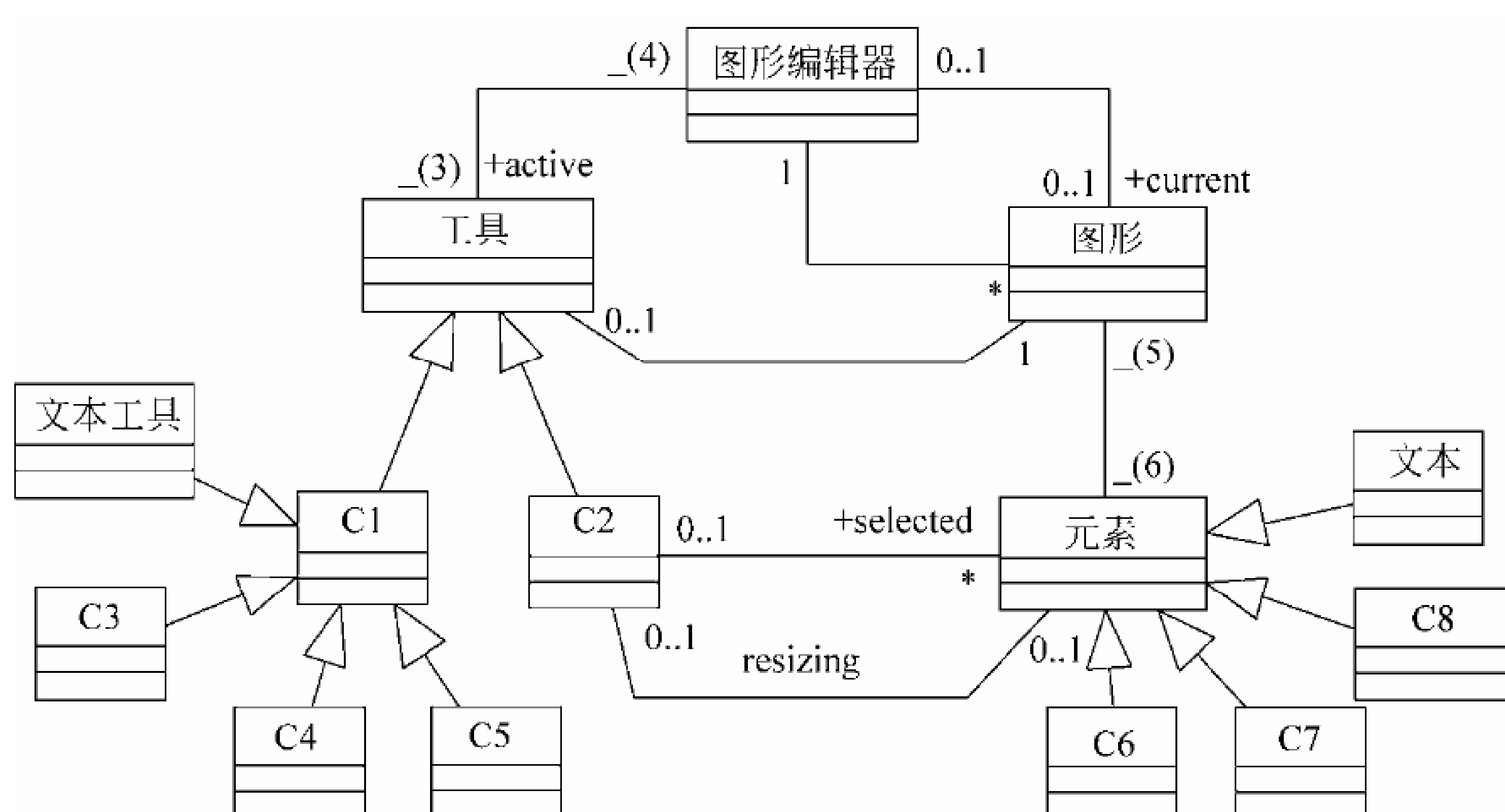


图 3-2 类图

【问题 1】（4 分）

根据说明中的描述，给出图 3-1 中 U1 和 U2 所对应的用例，以及（1）和（2）处所对应的关系。

【问题 2】（8 分）

根据说明中的描述，给出图 3-2 中缺少的 C1 至 C8 所对应的类名以及（3）至（6）处所对应的多重度。

【问题 3】（3 分）

图 3-2 中的类图设计采用了桥接（Bridge）设计模式，请说明该模式的内涵。

试题三分析

本题主要考查在面向对象分析与设计过程中，如何利用用例图和类图描述系统需求模型及设计模型。考生需要理解面向对象方法的相关概念和思想，并熟悉 UML 的语法及应用。用例图和类图是考试题中出现次数最多的两种 UML 模型。

【问题 1】

本问题考查用例之间的关系。在 UML 中，用例之间有 3 种关系：包含（include）、概括（generalize）和扩展（extend）。

如果多个用例中都含有相同的事件流，那么可以将其抽取出来放在一个单独的用例中，其他用例都可以通过包含（include）这个用例来使用其中的事件流。包含关系可以避免在多个用例的描述中重复拷贝相同的事件流。

概括关系是指子用例（child use case）继承父用例（parent use case）的行为，而子用例本身还可以增加新的行为或重置父类的某些行为。这种关系与面向对象程序设计中的“继承”很类似。

一个用例（基础用例，base use case）中加入一些新的动作后则构成了另外一个用例（扩展用例，extending use case），那么这两个用例之间的关系就是扩展关系。扩展关系与概括关系有相似之处，但是比概括关系更为严格。基础用例必须声明特定的扩展点，而扩展用例只能在这些扩展点上添加新行为。

由说明可知，图形编辑器的基本操作为创建图形、创建元素、选择元素和删除图形。对照图 3-1，可知这些最终都被确定为用例。除此之外，用例“创建图元”“创建文本”与用例“创建元素”之间是概括关系，即能创建的元素分别是图元和文本。图 3-1 中缺少了两个用例，而这两个用例都是与“选择元素”相关的。因此需要仔细阅读说明中关于“选择元素”的描述，其中最关键的一句描述为“拖曳选定元素的控制点，可以移动元素或者调整元素的大小”。这句话中出现了两个动词短语“移动元素”“调整元素大小”，这两个动作都是要先选择对应元素之后，才能实施的。因此，可以推出，U1 和 U2 应对应“移动元素”和“调整元素大小”。

下一步就是确定“移动元素”“调整元素大小”与“选择元素”之间的关系。由说明可知，必须先选择元素才能通过拖曳控制点来对元素进行相应的操作。因此，“移动元素”和“调整元素大小”是对“选择元素”的扩展，因此这三个用例之间应该是扩展关系。（1）和（2）处应填写 extend。

【问题 2】

本问题考查类图，考点是类层次结构及多重度。图 3-2 中有两个非常明显的继承结构，需要考生将其填充完整。这两个继承结构的最顶层父类分别是“工具”和“元素”，这就需要仔细阅读说明中与这两个词汇相关的描述。说明中第一次出现“工具”这个词，是在句子“编辑器提供了两种操作图形的工具：选择工具和创建工具”。这是典型的一般/特殊关系的描述，由此可以推断出，C1 和 C2 应该对应“选择工具”和“创建工具”，到底是怎样的对应关系，还需要进一步的细节信息。说明中的①给出“创建工具用于创建文本元素和图元元素”，而 C1 的一个子类就是“文本工具”，所以可以确定 C1 是“创建工具”，C2 是“选择工具”。那么 C3～C5 应该就是与创建图元元素相关的工具了，而图元分为三类：线条、矩形和椭圆。所以 C3～C5 分别对应“线条工具”“矩形工具”和“椭圆工具”。

现在图 3-2 中左边的继承结构已经填充完整了。右边的继承结构就可以对应地填写出来了，C6～C8 分别对应的是类“线条”“矩形”和“椭圆”。

确定多重度时，需要在说明寻找关联两端的类相关的描述。“对图形进行操作时，一次只能使用一种工具（即当前活动工具，active）”，即在图形编辑器中一次只能使用一个工具，而任何一个工具只属于这个图形编辑器。所以（3）处应填 0..1，（4）处应填 1。

一个图形可以包含多个元素，对于一个图形中的特定元素来说，只能属于这个图形。所以（5）处应填 1，（6）处应填 1..*。

【问题 3】

本问题考查桥接模式，该模式将抽象部分与其实现部分分离，使它们都可以独立地变化。

在以下情况中可以使用 Bridge 模式：

（1）不希望在抽象以及抽象的实现部分之间有一个固定的绑定关系。例如这种情况可能是因为，在程序运行时刻可以选择或切换实现部分。

（2）类的抽象以及它的实现都应该可以通过生成子类的方法加以扩充，使用 Bridge 模式可以对不同的抽象接口和实现部分进行组合，并分别对它们进行扩充。

（3）对一个抽象的实现部分的修改应该对用户不产生影响，即客户的代码不必重新编译。

参考答案

【问题 1】

U1：移动元素　　U2：调整元素大小（U1 和 U2 的答案可以互换）

（1）<<extend>>　　（2）<< extend>>

【问题 2】（8 分，C1～C8 每个 0.5 分，（3）～（6）每个 1 分）

C1：创建工具　　C2：选择工具　　C3：线条工具　　C4：矩形工具

C5：椭圆工具　　C6：线条　　C7：矩形　　C8：椭圆

注：C3～C5 的答案可以互换；C6~C8 的答案可以互换。

（3）0..1　　（4）1　　（5）1　　（6）1..* 或 *

【问题 3】

桥接模式将抽象部分与它的实现部分分离，使它们都可以独立地变化，对一个抽象的实现部分的修改应该对使用它的程序不产生影响。

试题四（共 15 分）

阅读下列说明和 C 代码，回答问题 1 至问题 3，将解答写在答题纸的对应栏内。

【说明】

某应用中需要对 100 000 个整数元素进行排序，每个元素的取值在 0～5 之间。排序算法的基本思想是：对每一个元素 *x*，确定小于等于 *x* 的元素个数（记为 *m*），将 *x* 放在输出元素序列的第 *m* 个位置。对于元素值重复的情况，依次放入第 *m*-1、*m*-2、…个位置。例如，如果元素值小于等于 4 的元素个数有 10 个，其中元素值等于 4 的元素个数有 3 个，则 4 应该在输出元素序列的第 10 个位置、第 9 个位置和第 8 个位置上。

算法具体的步骤为：

步骤 1：统计每个元素值的个数。

步骤 2：统计小于等于每个元素值的个数。

步骤 3：将输入元素序列中的每个元素放入有序的输出元素序列。

【C 代码】

下面是该排序算法的 C 语言实现。

（1）常量和变量说明

R：常量，定义元素取值范围中的取值个数，如上述应用中 R 值应取 6。

i：循环变量。

n：待排序元素个数。

a：输入数组，长度为 n。

b：输出数组，长度为 n。

c：辅助数组，长度为 R，其中每个元素表示小于等于下标所对应的元素值的个数。

（2）函数 sort

```
1 void sort(int n, int a[], int b[]) {
2    int c[R], i;
3    for(i = 0; i < ____(1)____; i++) {
4       c[i] = 0;
5    }
6    for(i = 0; i < n; i++) {
7        c[a[i]] = ____(2)____;
```

```
 8    }
 9    for(i = 1; i < R; i++) {
10        c[i] = ____(3)____;
11    }
12    for(i = 0; i < n; i++) {
13        b[c[a[i]] - 1] = ____(4)____;
14        c[a[i]] = c[a[i]] - 1;
15    }
16 }
```

【问题 1】（8 分）

根据说明和 C 代码，填充 C 代码中的空缺（1）～（4）。

【问题 2】（4 分）

根据 C 代码，函数的时间复杂度和空间复杂度分别为__（5）__和__（6）__（用 O 符号表示）。

【问题 3】（3 分）

根据以上 C 代码，分析该排序算法是否稳定。若稳定，请简要说明（不超过 100 字）；若不稳定，请修改其中代码使其稳定（给出要修改的行号和修改后的代码）。

试题四分析

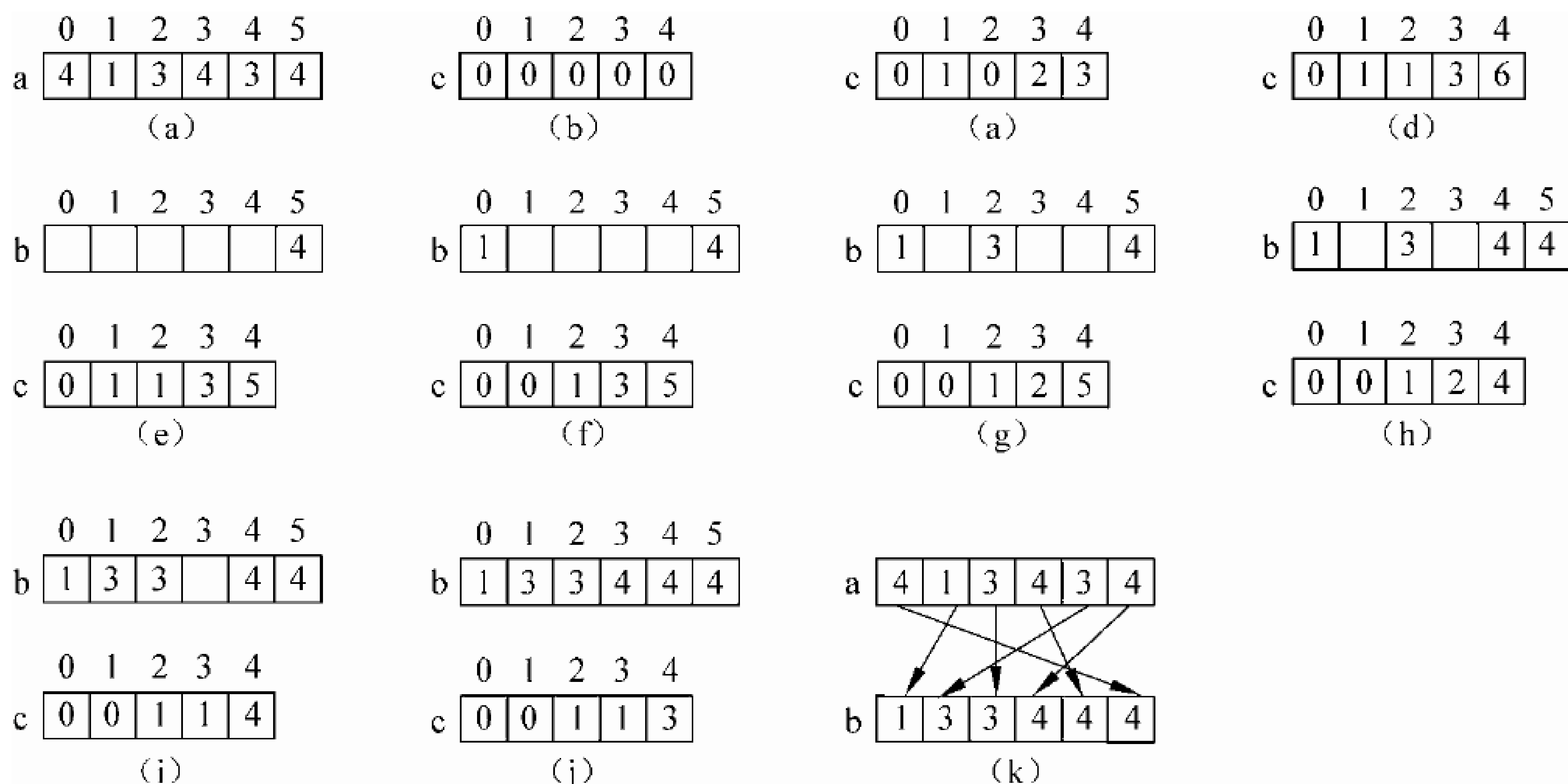
本题考查算法设计与分析技术以及算法的 C 语言实现，是比较传统的题目，要求考生细心分析题目中所描述的内容。

【问题 1】

根据题中说明，第 3 到第 5 行代码进行 c 数组的初始化，c 数组的长度为 R，在 C 语言中，下标从 0 开始，因此空格（1）中填写 R。第 6 到第 8 行检查 a 数组的每一个元素。如果元素的值为 i，则增加 c[i]的值。因此 c[a[i]] = c[a[i]] + 1，空格（2）填写 c[a[i]]+1。完成第 6 行到第 8 行的代码后，c[i]中就存放了等于 i 的元素的个数。第 9 到第 11 行，通过在数组 c 中记录计数和，c[i] = c[i - 1] + c[i]，可以确定对每一个 i=0,1,⋯,R-1，有多少个元素是小于或等于 i 的。因此空格（3）填写 c[i - 1] + c[i]。第 12 行到第 15 行把数组 a 中每个元素 a[i]放在输出数组 b 中与其相应的最终位置上，b[c[a[i]] - 1] =a[i]，因此空格（4）填写 a[i]。由于可能存在相同元素，因此每次将一个值 a[i]放入数组 b 中时，都要减小 c[i]的值。下面以一个例子来说明排序过程。

设 a = {4,1,3,4,3,4}，R = 5，即待排序的元素值在{0,1,2,3,4}中，其排序过程如下图所示。

图中（a）为输入数组 a，（b）为初始化后的 c 数组，（c）为统计数组 a 中每个元素后的 c 数组，（d）为计数和，即统计小于等于 i 的元素个数后的 c 数组。（e）到（j）是将 a 数组中的元素依次放到 b 数组的过程，（k）是数组 a 和数组 b 的元素对应关系。



【问题 2】

根据上述 C 代码，第 3 到第 5 行代码的 for 循环所花时间为 $\Theta(R)$。第 6 到第 8 行的 for 循环所花时间为 $\Theta(n)$。第 9 到第 11 行的 for 循环所花时间为 $\Theta(R)$。第 12 到第 15 行 for 循环所花时间为 $\Theta(n)$。因此整个算法的时间复杂度为 $\Theta(n+R)$。若 R 远小于 n 或者 $R=\Theta(n)$ 时，时间复杂度可以表示为 $\Theta(n)$。

【问题 3】

从图（k）可以看出，算法不稳定。算法不稳定的原因在于将数组 a 中元素放到数组 b 中时，是从数组 a 的第一个元素开始，依次取出元素放到数组 b 中。这样，相同的两个元素值，在数组 a 中的相对位置和在数组 b 中的相对位置正好相反。若从数组 a 的最后一个元素开始，依次向前取元素放到 b 数组中，可以保持相同元素的相对位置。因此将第 12 行的代码 for(i = 0; i < n;i++)改为 for(i = n - 1; i > 0;i--)，则排序算法是稳定的。

参考答案

【问题 1】

（1）R　　　　　　（2）c[a[i]] + 1

（3）c[i] + c[i - 1]　　　　（4）a[i]

【问题 2】

（5）$\Theta(n+R)$ 或者 $\Theta(n)$ 或 n 或线性

（6）$\Theta(n+R)$ 或者 $\Theta(n)$ 或 n 或线性

【问题 3】

不稳定。修改第 12 行的 for 循环为 for(i = n - 1; i >= 0; i--) 即可。

试题五（共 15 分）

阅读下列说明和 C++代码，将应填入＿（n）＿处的字句写在答题纸的对应栏内。

【说明】

某饭店在不同的时段提供多种不同的餐饮，其菜单的结构图如图 5-1 所示。

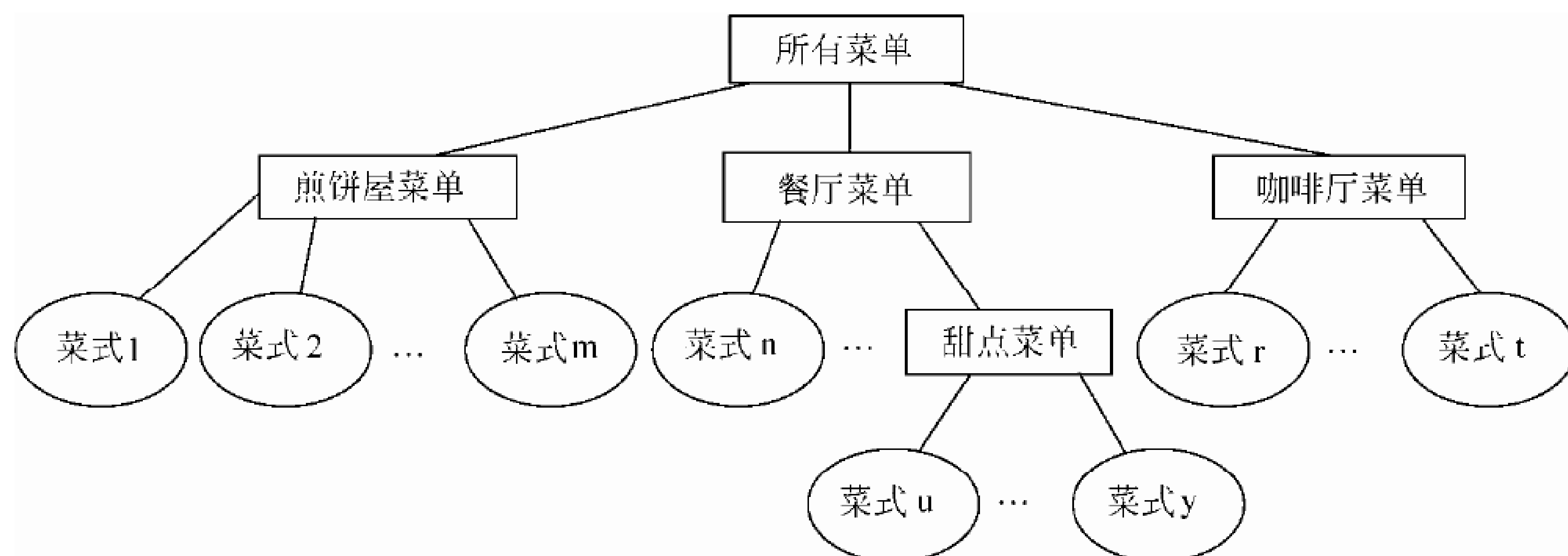


图 5-1 菜单结构图

现在采用组合（Composition）模式来构造该饭店的菜单，使得饭店可以方便地在其中增加新的餐饮形式，得到如图 5-2 所示的类图。其中 MenuComponent 为抽象类，定义了添加（add）新菜单和打印饭店所有菜单信息（print）的方法接口。类 Menu 表示饭店提供的每种餐饮形式的菜单，如煎饼屋菜单、咖啡屋菜单等。每种菜单中都可以添加子菜单，例如图 5-1 中的甜点菜单。类 MenuItem 表示菜单中的菜式。

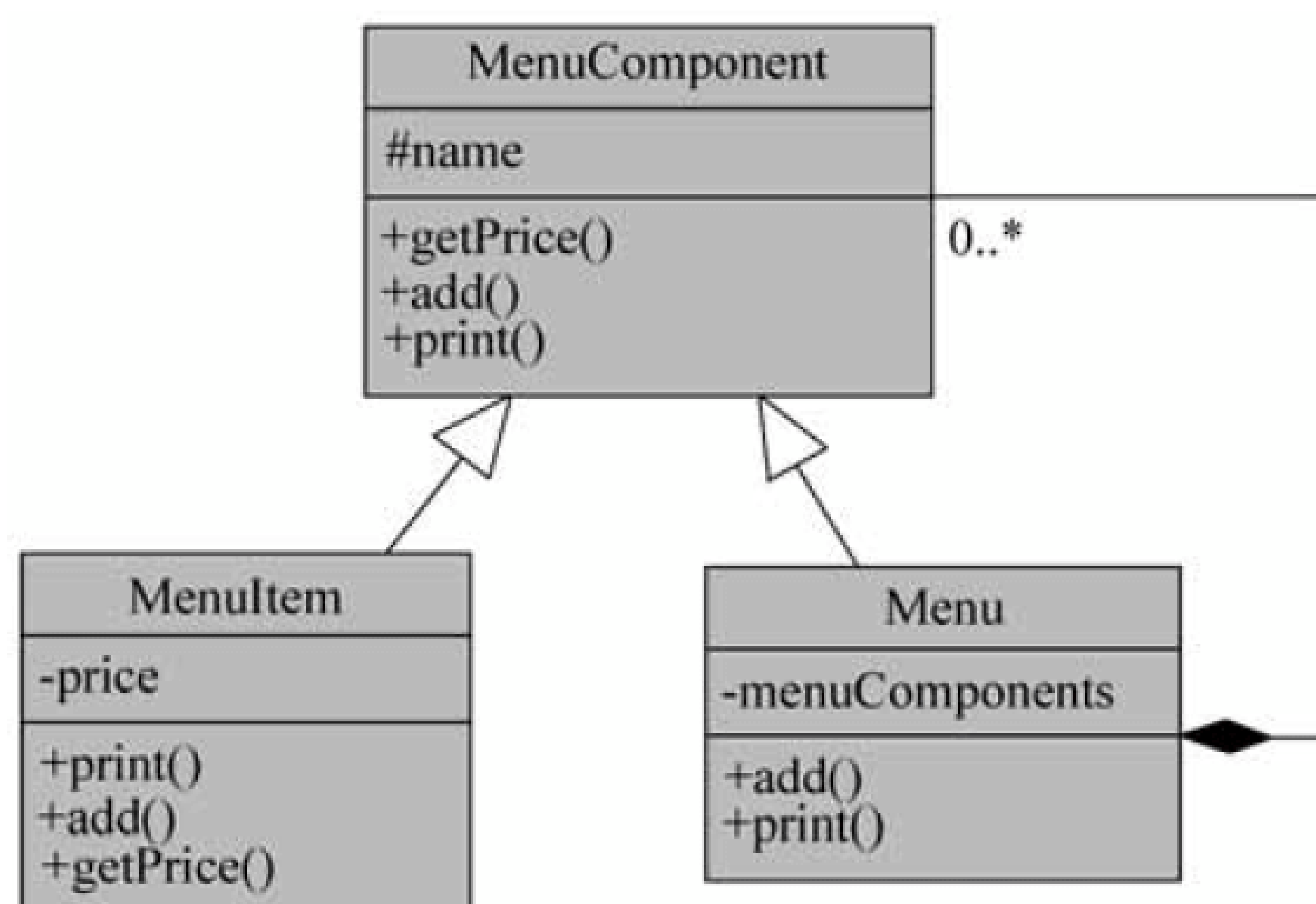


图 5-2 类图

【C++代码】

```
#include <iostream>
#include <list>
#include <string>
using namespace std;
class MenuComponent {
```

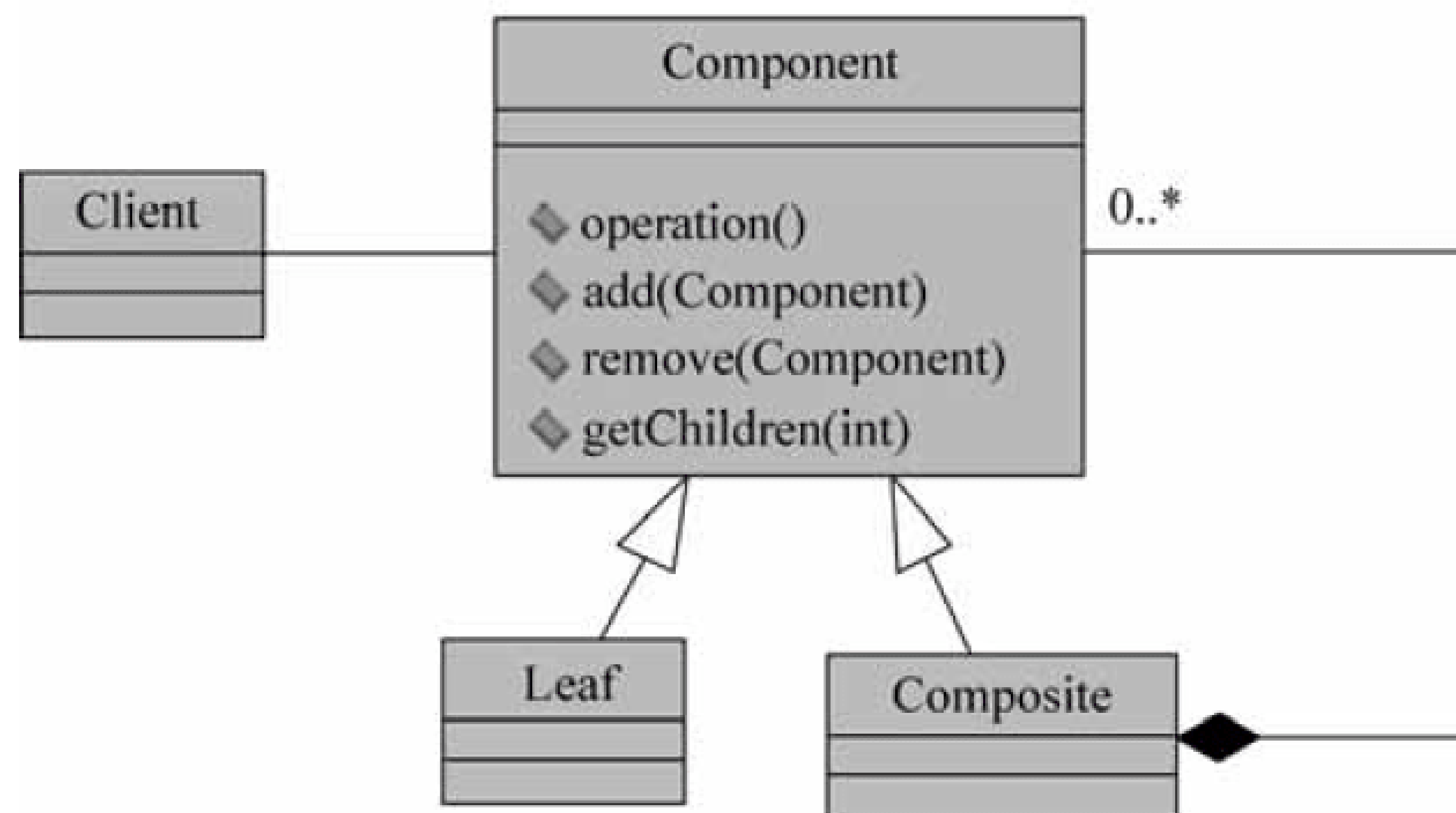
```
protected:  string name;
public:
   MenuComponent(string name) {  this->name = name; }
   string getName() { return name; }
   ___(1)___;                          // 添加新菜单
   virtual void print() = 0;           // 打印菜单信息
};
class MenuItem : public MenuComponent {
private:  double price;
public:
   MenuItem(string name, double price) : MenuComponent(name) { this->price
   = price; }
   double getPrice() { return price; }
   void add(MenuComponent* menuComponent) { return ; }   // 添加新菜单
   void print() { cout << "    " << getName() << ", " << getPrice() << endl; }
};
class Menu : public MenuComponent {
private:  list<___(2)___> menuComponents;
public:
   Menu(string name): MenuComponent(name){}
   void add(MenuComponent* menuComponent)                 // 添加新菜单
   {   ___(3)___;   }
   void print() {
     cout << "\n" << getName() << "\n----------------------------" << endl;
     std::list<MenuComponent *>::iterator iter;
     for(iter = menuComponents.begin(); iter != menuComponents.end(); iter++)
         ___(4)___->print();
   }
};
void main() {
  MenuComponent* allMenus = new Menu("ALL MENUS");
  MenuComponent* dinerMenu = new Menu("DINER MENU");
  …… // 创建更多的 Menu 对象，此处代码省略
  allMenus->add(dinerMenu);           // 将 dinerMenu 添加到餐厅菜单中
  …… // 为餐厅增加更多的菜单，此处代码省略
  ___(5)___->print();                 // 打印饭店所有菜单的信息
}
```

试题五分析

Composite 模式将对象组合成树形结构以表示“整体-部分”的层次结构，其中的组

合对象使得你可以组合基元对象以及其他的组合对象，从而形成任意复杂的结构。Composite 模式使得用户对单个对象和组合对象的使用具有一致性。

Composite 模式的结构下图所示。



其中：

- 类 Component 为组合中的对象声明接口，在适当的情况下，实现所有类共有接口的缺省行为，声明一个接口用于访问和管理 Component 的子部件；
- 类 Leaf 在组合中表示叶节点对象，叶节点没有子节点；并在组合中定义图元对象的行为；
- 类 Composite 定义有子部件的那些部件的行为，存储子部件，并在 Component 接口中实现与子部件有关的操作；
- 类 Client 通过 Component 接口操纵组合部件的对象。

下列情况可以使用 Composite 模式：

（1）表示对象的整体-部分层次结构；

（2）希望用户忽略组合对象与单个对象的不同，用户将统一地使用组合结构中的所有对象。

试题五将组合模式应用到饭店菜单的构造中。图 5-2 中的类 MenuComponent 对应上图中的 Component，MenuItem 对应 Leaf，Menu 对应 Composite。在实现时，通常都会把 Component 定义为抽象类。

在 C++中，抽象类是指至少包含一个纯虚拟函数的类。类 MenuComponent 中已经包含了一个纯虚拟函数 print，所以 MenuComponent 已经是一个抽象类了。（1）处根据注释，这里应该定义功能为“添加新菜单”的成员函数。在子类 MenuItem 和 Menu 中可以看到，都有 add 成员函数，说明子类中重置了父类中的成员函数。所以（1）处的成员函数也应该定义为纯虚拟函数，即 virtual void add(MenuComponent* menuComponent) = 0。

由图 5-2 可以看出，Menu 中包含了 MenuComponent 的对象集合。程序中用 C++标

准模板库中的 list 来实现这个聚集关系。因此（2）处应填入 MenuComponent *。由于使用了 list，就可以利用 list 中提供的各种方法了。list 中用于添加元素的方法是 push_back，所以（3）处应填入 menuComponents.push_back(menuComponent)。

（4）处出现在方法 print 中，其功能是打印出所有菜单的信息。这里使用了 list 中的迭代器类 iterator，遍历每个子菜单，并调用子菜单中定义的 print 方法打印该子菜单的信息。（4）处应填入*iter。

为了能够在 main 中打印出所有的菜单信息，必须使用表示菜单结构中最顶层菜单的对象来调用 print，因此（5）处应填入 allMenus。

参考答案

（1）virtual void add(MenuComponent* menuComponent) = 0

（2）MenuComponent *

（3）menuComponents.push_back(menuComponent)

（4）(*iter)

（5）allMenus

试题六（共 15 分）

阅读下列说明和 Java 代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

某饭店在不同的时段提供多种不同的餐饮，其菜单的结构图如图 6-1 所示。

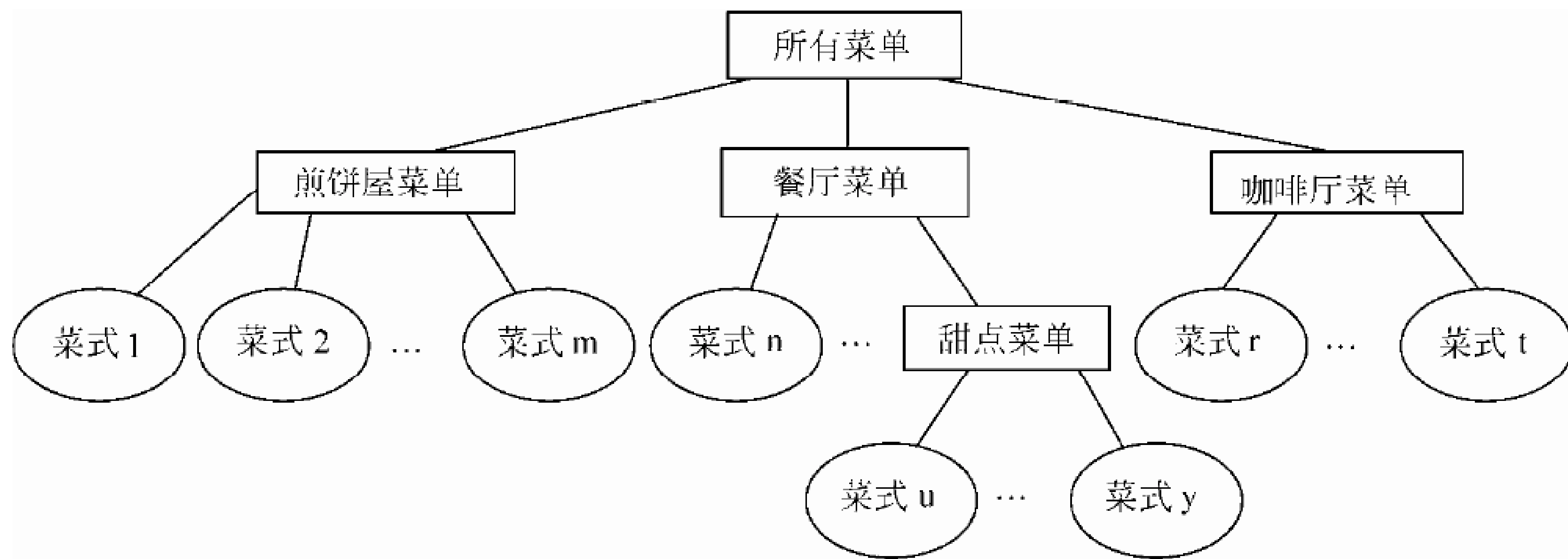


图 6-1　菜单结构图

现在采用组合（Composition）模式来构造该饭店的菜单，使得饭店可以方便地在其中增加新的餐饮形式，得到如图 6-2 所示的类图。其中 MenuComponent 为抽象类，定义了添加（add）新菜单和打印饭店所有菜单信息（print）的方法接口。类 Menu 表示饭店提供的每种餐饮形式的菜单，如煎饼屋菜单、咖啡屋菜单等。每种菜单中都可以添加子菜单，例如图 6-1 中的甜点菜单。类 MenuItem 表示菜单中的菜式。

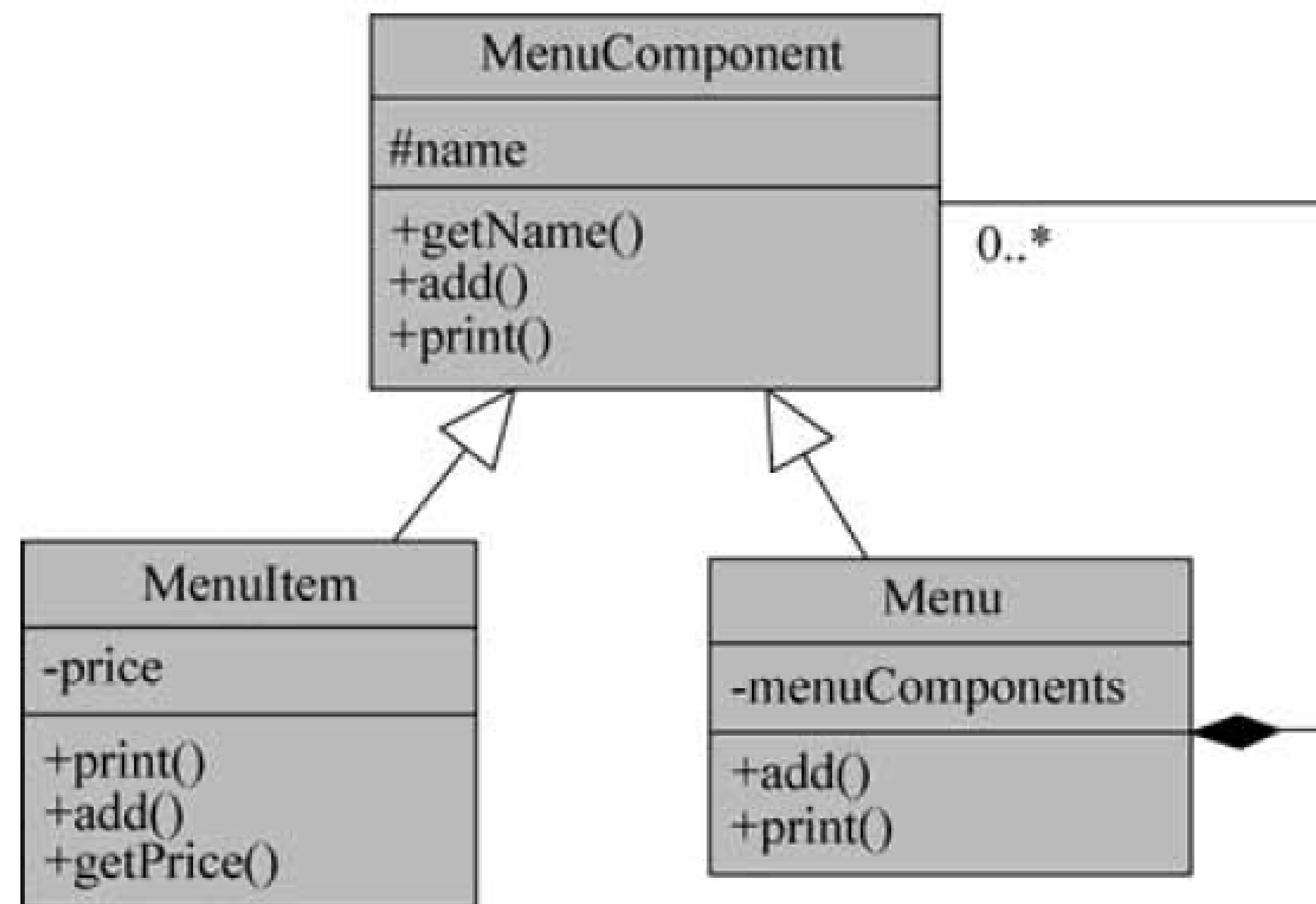


图 6-2　类图

【Java 代码】

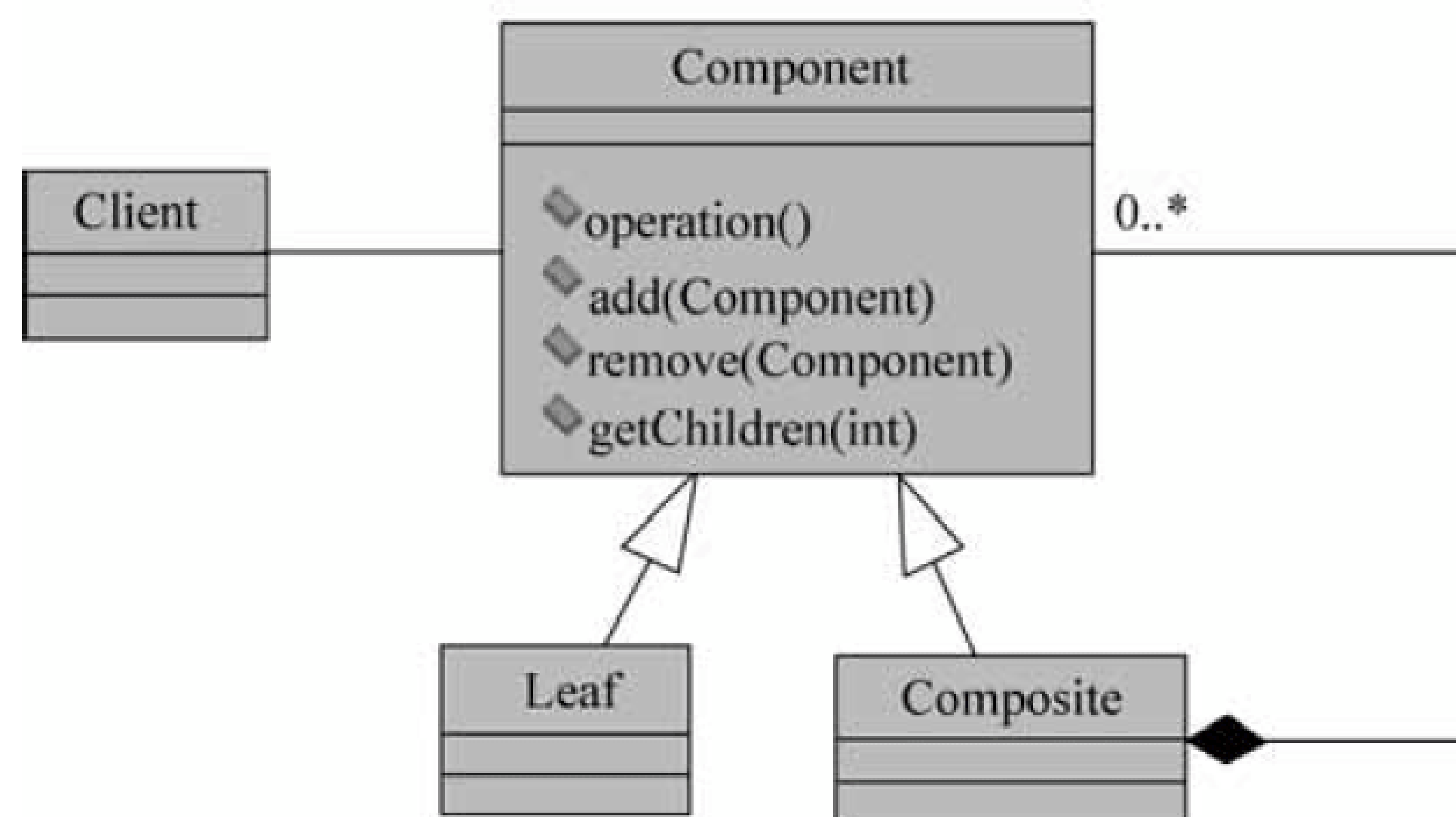
```
import java.util.*;
___(1)___ MenuComponent {
   protected String name;
   ___(2)___; //添加新菜单
   public abstract void print();                    //打印菜单信息
   public String getName() { return name; }
}
class MenuItem extends MenuComponent {
   private double price;
   public MenuItem(String name, double price) {
        this.name = name;   this.price = price;
   }
   public double getPrice()  { return price; }
   public void add(MenuComponent menuComponent){ return ; } // 添加新菜单
   public void print() {
     System.out.print(" " + getName());
     System.out.println("," + getPrice());
   }
}
class Menu extends MenuComponent {
   private List<MenuComponent> menuComponents = new ArrayList<Menu-
   Component>();
   public Menu(String name) { this.name = name; }
   public void add(MenuComponent menuComponent) {          //添加新菜单
     menuComponents.___(3)___;
   }
```

```
    public void print() {
      System.out.print("\n" + getName());
      System.out.println("," + "---------------------");
      Iterator iterator = menuComponents.iterator();
      while(iterator.hasNext()) {
        MenuComponent menuComponent = (MenuComponent)iterator.next();
        ___(4)___;
      }
    }
  }
  class MenuTestDrive {
    public static void main(String args[]) {
        MenuComponent  allMenus = new Menu("ALL MENUS");
  MenuComponent  dinerMenu = new Menu("DINER MENU");
        ...... // 创建更多的 Menu 对象，此处代码省略
        allMenus.add(dinerMenu);    // 将 dinerMenu 添加到餐厅菜单中
        ...... // 为餐厅增加更多的菜单，此处代码省略
        ___(5)___;    // 打印饭店所有菜单的信息
    }
  }
```

试题六分析

Composite 模式将对象组合成树形结构以表示“整体-部分”的层次结构，其中的组合对象使得你可以组合基元对象以及其他的组合对象，从而形成任意复杂的结构。Composite 模式使得用户对单个对象和组合对象的使用具有一致性。

Composite 模式的结构如下图所示。



其中：

- 类 Component 为组合中的对象声明接口，在适当的情况下，实现所有类共有接口的缺省行为，声明一个接口用于访问和管理 Component 的子部件；

- 类 Leaf 在组合中表示叶节点对象，叶节点没有子节点；并在组合中定义图元对象的行为；
- 类 Composite 定义有子部件的那些部件的行为，存储子部件，并在 Component 接口中实现与子部件有关的操作；
- 类 Client 通过 Component 接口操纵组合部件的对象。

下列情况可以使用 Composite 模式：

（1）表示对象的整体-部分层次结构；

（2）希望用户忽略组合对象与单个对象的不同，用户将统一地使用组合结构中的所有对象。

试题六将组合模式应用到饭店菜单的构造中。图 6-2 中的类 MenuComponent 对应上图中的 Component，MenuItem 对应 Leaf，Menu 对应 Composite。在实现时，通常都会把 Component 定义为抽象类。

在 Java 中，用 abstract 关键字限定的类即为抽象类，所以（1）处应填入 abstract class。（2）处根据注释，这里应该定义功能为“添加新菜单”的成员函数。在子类 MenuItem 和 Menu 中可以看到，都有 add 成员函数，说明子类中重置了父类中的成员函数。所以（2）处应填入 public abstract void add(MenuComponent menuComponent)。

由图 6-2 可以看出，Menu 中包含了 MenuComponent 的对象集合。程序中用 Java 中的 list 来实现这个聚集关系，这样就可以利用 list 中提供的各种方法了。list 中用于添加元素的方法是 add，所以（3）处应填入 add(menuComponent)。

（4）处出现在方法 print 中，其功能是打印出所有菜单的信息。这里使用了 list 中的迭代器类 iterator，遍历每个子菜单，并调用子菜单中定义的 print 方法打印该子菜单的信息。（4）处应填入 menuComponent.print()。

为了能够在 main 中打印出所有的菜单信息，必须使用表示菜单结构中最顶层菜单的对象来调用 print，因此（5）处应填入 allMenus.print()。

参考答案

（1）abstract class 或 public abstract class

（2）public abstract void add(MenuComponent menuComponent)

或 abstract void add(MenuComponent menuComponent)

或 protected abstract void add(MenuComponent menuComponent)

（3）add(menuComponent)

（4）menuComponent.print()

（5）allMenus.print()

第 11 章　2011 下半年软件设计师上午试题分析与解答

试题（1）

若某条无条件转移汇编指令采用直接寻址，则该指令的功能是将指令中的地址码送入__（1）__。

（1）A．PC（程序计数器）　　B．AR（地址寄存器）
　　C．AC（累加器）　　D．ALU（算逻运算单元）

试题（1）分析

本题考查指令系统基础知识。

直接寻址是指操作数存放在内存单元中，指令中直接给出操作数所在存储单元的地址。而跳转指令中的操作数即为要转向执行的指令地址，因此，应将指令中的地址码送入程序计数器（PC），以获得下一条指令的地址，从而实现程序执行过程的自动控制功能。

参考答案

（1）A

试题（2）

若某计算机系统的 I/O 接口与主存采用统一编址，则输入输出操作是通过__（2）__指令来完成的。

（2）A．控制　　B．中断　　C．输入输出　　D．访存

试题（2）分析

本题考查计算机系统中的输入输出系统基础知识。

常用的 I/O 接口编址方法有两种：一是与内存单元统一编址，二是单独编址。

与内存单元统一编址方式下，是将 I/O 接口中有关的寄存器或存储部件看作存储器单元，与主存中的存储单元统一编址。这样，内存地址和接口地址统一在一个公共的地址空间里，对 I/O 接口的访问就如同对主存单元的访问一样，可以用访问内存单元的指令访问 I/O 接口。

I/O 接口单独编址是指通过设置单独的 I/O 地址空间，为接口中的有关寄存器或存储部件分配地址码，需要设置专门的 I/O 指令进行访问。这种编址方式的优点是不占用主存的地址空间，访问主存的指令和访问接口的指令不同，在程序中容易使用和辨认。

参考答案

（2）D

试题（3）

在程序的执行过程中，Cache 与主存的地址映像由__（3）__。

（3）A．专门的硬件自动完成　　B．程序员进行调度
　　C．操作系统进行管理　　D．程序员和操作系统共同协调完成

试题（3）分析

本题考查存储系统基础知识。

高速缓存（Cache）的出现主要有两个因素：首先是由于 CPU 的速度和性能提高很快而主存速度较低且价格高，其次就是程序执行的局部性特点。因此，才将速度比较快而容量有限的静态存储器芯片构成 Cache，以尽可能发挥 CPU 的高速度。因此，必须用硬件来实现 Cache 的全部功能。

参考答案

（3）A

试题（4）

总线复用方式可以__（4）__。

（4）A．提高总线的传输带宽　　B．增加总线的功能
　　C．减少总线中信号线的数量　　D．提高 CPU 利用率

试题（4）分析

本题考查总线基础知识。

总线是一组能为多个部件分时共享的信息传送线，用来连接多个部件并为之提供信息交换通路，通过总线复用方式可以减少总线中信号线的数量，以较少的信号线传输更多的信息。

参考答案

（4）C

试题（5）

在 CPU 的寄存器中，__（5）__对用户是完全透明的。

（5）A．程序计数器　　B．指令寄存器　　C．状态寄存器　　D．通用寄存器

试题（5）分析

本题考查计算机系统基础知识。

寄存器组是 CPU 中的一个重要组成部分，它是 CPU 内部的临时存储空间。寄存器既可以用来存放数据和地址，也可以存放控制信息或 CPU 工作时的状态。在 CPU 中增加寄存器的数量，可以使 CPU 把执行程序时所需的数据尽可能地放在寄存器中，从而减少访问内存的次数，提高其运行速度。但是，寄存器的数目也不能太多，除了增加成本外，寄存器地址编码增加还会增加指令的长度。CPU 中的寄存器通常分为存放数据的寄存器、存放地址的寄存器、存放控制信息的寄存器、存放状态信息的寄存器和其他寄存器等类型。

程序计数器是存放指令地址的寄存器，其作用是：当程序顺序执行时，每取出一条指令，程序计数器（PC）内容自动增加一个值，指向下一条要取的指令。当程序出现转移时，则将转移地址送入 PC，然后由 PC 指向新的指令地址。

指令寄存器（IR）用于存放正在执行的指令，指令从内存取出后送入指令寄存器。其操作码部分经指令译码器送微操作信号发生器，其地址码部分指明参加运算的操作数的地址形成方式。在指令执行过程中，指令寄存器中的内容保持不变。

状态字寄存器（PSW）用于保存指令执行完成后产生的条件码，例如运算是否有溢出，结果为正还是为负，是否有进位等。此外，PSW 还保存中断和系统工作状态等信息。

通用寄存器组是 CPU 中的一组工作寄存器，运算时用于暂存操作数或地址。在程序中使用通用寄存器可以减少访问内存的次数，提高运算速度。

在汇编语言程序中，程序员可以直接访问通用寄存器以存取数据，可以访问状态字寄存器以获取有关数据处理结果的相关信息，可以通过相对程序计数器进行寻址，但是不能访问指令寄存器。

参考答案

（5）B

试题（6）

CPU 中译码器的主要作用是进行__（6）__。

（6）A．地址译码　　B．指令译码
　　C．数据译码　　D．选择多路数据至 ALU

试题（6）分析

本题考查计算机系统基础知识。

CPU 中指令译码器的功能是对现行指令进行分析，确定指令类型和指令所要完成的操作以及寻址方式，并将相应的控制命令发往相关部件。

参考答案

（6）B

试题（7）

利用__（7）__可以获取某 FTP 服务器中是否存在可写目录的信息。

（7）A．防火墙系统　　B．漏洞扫描系统
　　C．入侵检测系统　　D．病毒防御系统

试题（7）分析

本题考查网络安全方面网络攻击和防御相关的基础知识。

漏洞扫描技术是检测远程或本地系统安全脆弱性的一种安全技术。通过与目标主机 TCP/IP 端口建立连接并请求某些服务（如 TELNET、FTP 等），记录目标主机的应答，搜集目标主机相关信息（如匿名用户是否可以登录等），从而发现目标主机某些内在的安全弱点。

参考答案

（7）B

试题（8）

通过内部发起连接与外部主机建立联系，由外部主机控制并盗取用户信息的恶意代码为__（8）__。

（8）A．特洛伊木马 B．蠕虫病毒 C．宏病毒 D．CIH 病毒

试题（8）分析

本题考查网络安全方面病毒相关的基础知识。

典型网络病毒主要有宏病毒、特洛伊木马、蠕虫病毒、脚本语言病毒等。

宏病毒的传播方式通常如下：字处理程序 Word 在打开一个带宏病毒的文档或模板时，激活了病毒宏，病毒宏将自身复制至 Word 的通用（Normal）模板中，以后在打开或关闭文件时病毒宏就会把病毒复制到该文件中。

特洛伊木马是一种秘密潜伏且能够通过远程网络进行控制的恶意程序。控制者可以控制被秘密植入木马的计算机的一切动作和资源，是恶意攻击者窃取信息的工具。

蠕虫病毒的传播过程一般表现为：蠕虫程序驻于一台或多台机器中，它会扫描其他机器是否有感染同种计算机蠕虫，如果没有，就会通过其内建的传播手段进行感染，以达到使计算机瘫痪的目的。

参考答案

（8）A

试题（9）

从认证中心 CA 获取用户 B 的数字证书，该证书用__（9）__作数字签名；从用户 B 的数字证书中可以获得 B 的公钥。

（9）A．CA 的公钥 B．CA 的私钥 C．B 的公钥 D．B 的私钥

试题（9）分析

本题考查数字证书的基础知识。

用户的数字证书由某个可信的证书发放机构（Certification Authority，CA）建立，并由 CA 或用户将其放入公共目录中。在 X.509 标准中，一般格式的数字证书包含以下数据域：

- 版本号：用于区分 X.509 的不同版本
- 序列号：由同一发行者（CA）发放的每个证书的序列号是唯一的
- 签名算法：签署证书所用的算法及其参数
- 发行者：指建立和签署证书的 CA 的 X.509 名字
- 有效期：包括证书有效期的起始时间和终止时间
- 主体名：指证书持有者的名称及有关信息
- 公钥：证书持有者的公钥以及其使用方法

- 发行者 ID：任选的名字唯一地标识证书的发行者
- 主体 ID：任选的名字唯一地标识证书的持有者
- 扩展域：添加的扩充信息
- 认证机构的签名：用 CA 私钥对证书的签名

从上述描述可知，数字证书用 CA 私钥做数字签名，从用户的数字证书中可以获得用户的公钥。

参考答案

（9）B

试题（10）

（10）指可以不经著作权人许可，不需支付报酬，使用其作品。

（10）A．合理使用　　B．许可使用

C．强制许可使用　　D．法定许可使用

试题（10）分析

本题考查知识产权方面的基础知识。

合理使用是指在特定的条件下，法律允许他人自由使用享有著作权的作品而不必征得著作权人的同意，也不必向著作权人支付报酬，但应当在指明著作权人姓名、作品名称，并且不侵犯著作权人依法享有的合法权利的情况下对著作权人的作品进行使用。

许可使用是指著作权人将自己的作品以一定的方式、在一定的地域和期限内许可他人使用，并由此获得经济利益。

强制许可使用是指在一定条件下，作品的使用者基于某种正当理由，需要使用他人已发表的作品，经申请由著作权行政管理部门授权即可使用该作品，无须征得著作权人同意，但应向其支付报酬。

法定许可是指除著作权人声明不得使用外，使用人在未经著作权人许可的情况下，向著作权人支付报酬，指明著作权人姓名、作品名称，并且不侵犯著作权人依法享有的合法权利的情况下进行使用。

参考答案

（10）A

试题（11）

王某是 M 国际运输有限公司计算机系统管理员。任职期间，王某根据公司的业务要求开发了“海运出口业务系统”，并由公司使用，随后，王某向国家版权局申请了计算机软件著作权登记，并取得了《计算机软件著作权登记证书》。证书明确软件名称是“海运出口业务系统 V1.0”，著作权人为王某。以下说法中，正确的是（11）。

（11）A．海运出口业务系统 V1.0 的著作权属于王某

B．海运出口业务系统 V1.0 的著作权属于 M 公司

C．海运出口业务系统 V1.0 的著作权属于王某和 M 公司

D. 王某获取的软件著作权登记证是不可以撤销的

试题（11）分析

本题考查知识产权方面的基础知识。

王某开发的软件（即“海运出口业务系统 V1.0”）是在国际运输有限公司担任计算机系统管理员期间根据国际运输有限公司业务要求开发的，该软件是针对本职工作中明确指定的开发目标所开发的。根据《著作权法》第 16 条规定，公民为完成法人或者非法人单位工作任务所创作的作品是职务作品。认定作品为职务作品还是个人作品，应考虑两个前提条件：一是作者和所在单位存在劳动关系，二是作品的创作属于作者应当履行的职责。职务作品分为一般职务作品和特殊的职务作品：一般职务作品的著作权由作者享有，单位或其他组织享有在其业务范围内优先使用的权利，期限为二年；特殊的职务作品，除署名权以外，著作权的其他权利由单位享有。所谓特殊职务作品是指《著作权法》第 16 条第 2 款规定的两种情况：一是主要利用法人或者其他组织的物质技术条件创作，并由法人或者其他组织承担责任的工程设计、产品设计图、计算机软件、地图等科学技术作品；二是法律、法规规定或合同约定著作权由单位享有的职务作品。《计算机软件保护条例》也有类似的规定，在第十三条中规定了三种情况，一是针对本职工作中明确指定的开发目标所开发的软件；二是开发的软件是从事本职工作活动所预见的结果或者自然的结果；三是主要使用了法人或者其他组织的资金、专用设备、未公开的专门信息等物质技术条件所开发并由法人或者其他组织承担责任的软件。王某在公司任职期间利用公司的资金、设备和各种资料，且是从事本职工作活动所预见的结果。所以，其进行的软件开发行为是职务行为(只要满足上述三个条件之一),其工作成果应由公司享有。因此，该软件的著作权应属于国际运输有限公司，但根据法律规定，王某享有署名权。

根据《计算机软件保护条例》第 7 条规定，软件登记机构发放的登记证明文件是登记事项的初步证明，只是证明登记主体享有软件著作权以及订立许可合同、转让合同的重要的书面证据，并不是软件著作权产生的依据。因为，软件著作权是自软件开发完成之日起自动产生的，未经登记的软件著作权或软件著作权专有合同和转让合同仍受法律保护。因此，软件登记机构发放的登记证明并不是软件著作权最终归属的证明，如果有相反证明，软件著作权登记证是可以撤销的。该软件是王某针对本职工作中明确指定的开发目标所开发的，该软件的著作权应属于公司。明确真正的著作权人之后，软件著作权登记证书的证明力自然就消失了（只有审判机关才能确定登记证书的有效性）。

参考答案

（11）B

试题（12）

计算机通过 MIC（话筒接口）收到的信号是__（12）__。

（12）A. 音频数字信号　　B. 音频模拟信号
　　C. 采样信号　　D. 量化信号

试题（12）分析

本题考查多媒体基础知识。

MIC（话筒）输出的是音频模拟信号，声卡从 MIC 获取音频模拟信号后，通过模数转换器（ADC），将声波振幅信号采样转换成一串数字信号并存储到计算机中。重放时，这些数字信号送到数模转换器（DAC），以同样的采样速度还原为模拟波形，放大后送到扬声器发声，这一技术称为脉冲编码调制技术（PCM）。

参考答案

（12）B

试题（13）

<u>（13）</u>既不是图像编码也不是视频编码的国际标准。

（13）A．JPEG　　B．MPEG　　C．ADPCM　　D．H.261

试题（13）分析

本题考查多媒体基础知识。

计算机中使用的图像压缩编码方法有多种国际标准和工业标准。目前广泛使用的编码及压缩标准有 JPEG、MPEG 和 H.261。

JEPG（Joint Photographic Experts Group）是一个由 ISO 和 IEC 两个组织机构（国际标准化组织）联合组成的一个专家组，负责制定静态和数字图像数据压缩编码标准，这个专家组开发的算法称为 JPEG 算法，并且成为国际上通用的标准。JPEG 是一个适用范围很广的静态图像数据压缩标准，既可用于灰度图像又可用于彩色图像。

MPEG（Moving Pictures Experts Group）动态图像压缩标准是一个由 ISO 和 IEC 两个组织机构联合组成的一个活动图像专家组制定的标准。1992 年提出 MPEG-1、MPEG-2 标准，用于实现全屏幕压缩编码及解码。MPEG-1 是针对传输率为 1Mbps 到 1.5Mbps 的普通电视质量的视频信号的压缩，MPEG-2 是对每秒 30 帧的 720×572 分辨率的视频信号进行压缩。1999 年发布了 MPEG-4 多媒体应用标准，目前推出了 MPEG-7 多媒体内容描述接口标准等。每个新标准的产生都极大地推动了数字视频的发展和更广泛的应用。

H.261 视频通信编码标准是由国际电话电报咨询委员会 CCITT（Consultative Committee on International Telephone and Telegraph）于 1998 年提出的电话/会议电视的建议标准，该标准又称为 P×64K 标准。CCITT 推出的 H.263 标准用于低位速率通信的电视图像编码。

参考答案

（13）C

试题（14）

多媒体制作过程中，不同媒体类型的数据收集、制作需要不同的软、硬件设备和技术手段，动画制作一般通过<u>（14）</u>进行。

（14）A．字处理软件　　B．视频卡

C．声卡　　　　　　　　　　D．图形/图像软件

试题（14）分析

本题考查多媒体基础知识。

在多媒体应用中，很重要的一个环节是制作所需要的各种媒体素材。声卡用于处理音频信息，它可以把话筒、录音机、电子乐器等输入的声音信息进行模数转换（A/D）、压缩等处理，也可以把经过计算机处理的数字化的声音信号通过还原（解压缩）、数模转换（D/A）后用音箱播放出来，或者用录音设备记录下来。音频卡的关键技术包括数字音频、音乐合成（FM 合成和波形表合面）、MIDI（乐器数字接口）和音效。数字音频部分具有 44.1kHz 的采样率，8 位以上的分辨率；具有录音和播放声音信号的功能；同时具有压缩采样信号的能力，最常用的压缩方法是自适应脉冲编码调制。数字音频的实现有不同的方法和芯片，大多数采用的是 CODEC 芯片，它具有硬件压缩功能。

视频卡是基于 PC 的一种多媒体视频信号处理设备，用来支持视频信号的输入与输出。它可以采集视频源、声频源和激光视盘机、录像机、摄像机等设备的信息，经过编辑或特技等处理而产生非常精美的画面。还可以对这些画面进行捕捉、数字化、冻结、存储、压缩、输出等操作。对画面的修整、像素显示调整、缩放功能等都是视频卡支持的标准功能。视频卡的功能是连接摄像机、VCR 影碟机、TV 等设备，以便获取、处理和表现各种动画和数字化视频媒体。

多媒体素材编辑软件用于采集、整理和编辑各种媒体数据。

文字处理软件的功能主要是文字处理，包括文字录入、编辑，文档编辑、排版、管理、打印、表格处理等功能，使用文字处理软件可以创建符合用户要求的、美观的文稿。常用的字处理软件有 WPS、Word、Notebook（记事本）、Writer（写字板）等。

图形/图像软件的主要功能包括显示和编辑图形/图像、图像压缩、图像捕捉、图形/图像素材库制作等。例如，Photoshop 用于图像的设计、编辑与处理，其功能强大，是使用最多的一种图形/图像工具软件；Xara3D 是一种 3D 图形软件，可用于制作高质量的三维动画。

参考答案

（14）D

试题（15）

确定软件的模块划分及模块之间的调用关系是__（15）__阶段的任务。

（15）A．需求分析　　B．概要设计　　C．详细设计　　D．编码

试题（15）分析

本题考查软件开发过程和软件开发阶段的基础知识。

需求分析确定软件要完成的功能及非功能性要求；概要设计将需求转化为软件的模块划分，确定模块之间的调用关系；详细设计将模块进行细化，得到详细的数据结构和算法；编码根据详细设计进行代码的编写，得到可以运行的软件，并进行单元测试。

参考答案

（15）B

试题（16）

利用结构化分析模型进行接口设计时，应以__（16）__为依据。

（16）A．数据流图　B．实体—关系图　C．数据字典　D．状态—迁移图

试题（16）分析

本题考查结构化分析与设计基础知识。

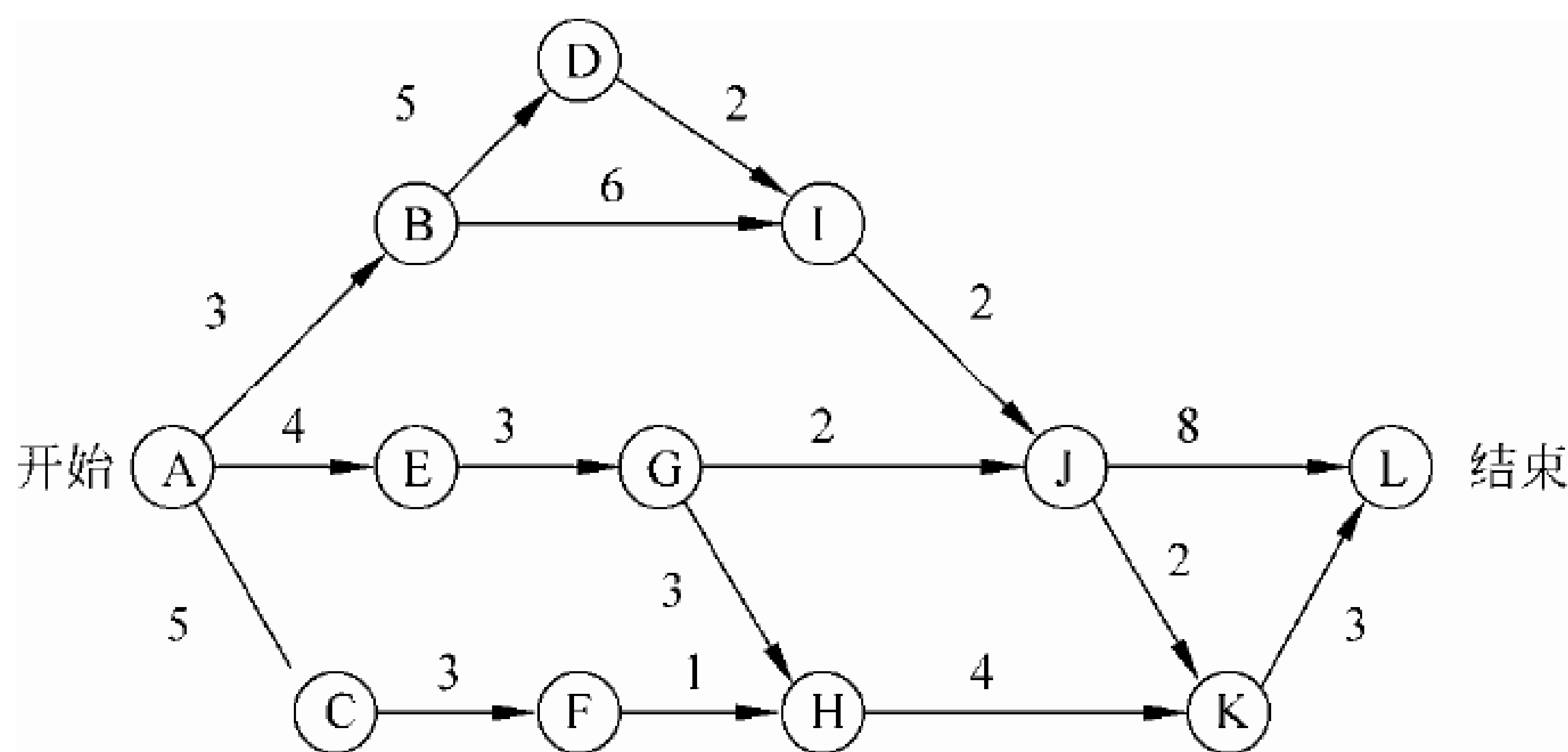
软件设计必须依据软件的需求来进行，结构化分析的结果为结构化设计提供了最基本的输入信息，其关系为：根据加工规格说明和控制规格说明进行过程设计；根据数据字典和实体关系图进行数据设计；根据数据流图进行接口设计；根据数据流图进行体系结构设计。

参考答案

（16）A

试题（17）

下图是一个软件项目的活动图，其中顶点表示项目里程碑，连接顶点的边表示包含的活动，边上的值表示完成活动所需要的时间，则关键路径长度为__（17）__。



（17）A．20　B．19　C．17　D．16

试题（17）分析

本题考查软件项目管理的相关知识。

关键路径是从开始到结束的最长路径，也是完成项目所需要的最短时间。根据上述活动图，路径 A-B-D-I-J-L 是关键路径，其长度为 20。

参考答案

（17）A

试题（18）

甘特图（Gantt 图）不能__（18）__。

（18）A．作为项目进度管理的一个工具

B．清晰地描述每个任务的开始和截止时间

C．清晰地获得任务并行进行的信息

D．清晰地获得各任务之间的依赖关系

试题（18）分析

本题考查软件项目管理的相关知识。

甘特图（Gantt 图）是进行项目进度管理的一个重要工具，它对项目进度进行描述，显示在什么地方活动是并行进行的，并用颜色或图标来指明完成的程度。使用该图，项目经理可以清晰的了解每个任务的开始和截止时间，哪些任务可以并行进行，哪些在关键路径上，但是不能很清晰的看出各任务之间的依赖关系。

参考答案

（18）D

试题（19）

以下关于风险管理的叙述中，不正确的是__（19）__。

（19）A．仅根据风险产生的后果来对风险排优先级

B．可以通过改变系统性能或功能需求来避免某些风险

C．不可能去除所有风险，但可以通过采取行动来降低或者减轻风险

D．在项目开发过程中，需要定期地评估和管理风险

试题（19）分析

本题考查风险管理知识。

风险是一种具有负面后果的、人们不希望发生的事件。项目经理必须进行风险管理，以了解和控制项目中的风险。

风险可能会发生，因此具有一定的概率；风险产生的后果严重程度不一样，因此需要区分。在对风险进行优先级排序时，需要根据风险概率和后果来进行排序。在确定了风险之后，根据实际情况，可以通过改变系统的性能或功能需求来避免某些风险。在项目开发过程中，不可能去除所有风险，但是可以通过采取行动来降低或者减轻风险。而且风险需要定期地评估和管理。

参考答案

（19）A

试题（20）

若 C 程序的表达式中引用了未赋初值的变量，则__（20）__。

（20）A．编译时一定会报告错误信息，该程序不能运行

B．可以通过编译并运行，但运行时一定会报告异常

C．可以通过编译，但链接时一定会报告错误信息而不能运行

D．可以通过编译并运行，但运行结果不一定是期望的结果

试题（20）分析

本题考查程序语言翻译基础知识。

在编写 C/C++源程序时，为所定义的变量赋初始值是良好的编程习惯，而赋初值不是强制的要求，因此编译程序不检查变量是否赋初值。如果表达式中引用的变量从定义到使用始终没有赋值，则该变量中的值表现为一个随机数，这样对表达式的求值结果就是不确定的了。

参考答案

（20）D

试题（21）

若二维数组 arr[1..M, 1..N]的首地址为 base，数组元素按列存储且每个元素占用 K 个存储单元，则元素 arr[i, j]在该数组空间的地址为__（21）__。

（21）A．base+((i–1)*M+j–1)*K　　B．base+((i–1)*N+j–1)*K

C．base+((j–1)*M+i–1)*K　　D．base+((j–1)*N+i–1)*K

试题（21）分析

本题考查数组元素的存储知识。

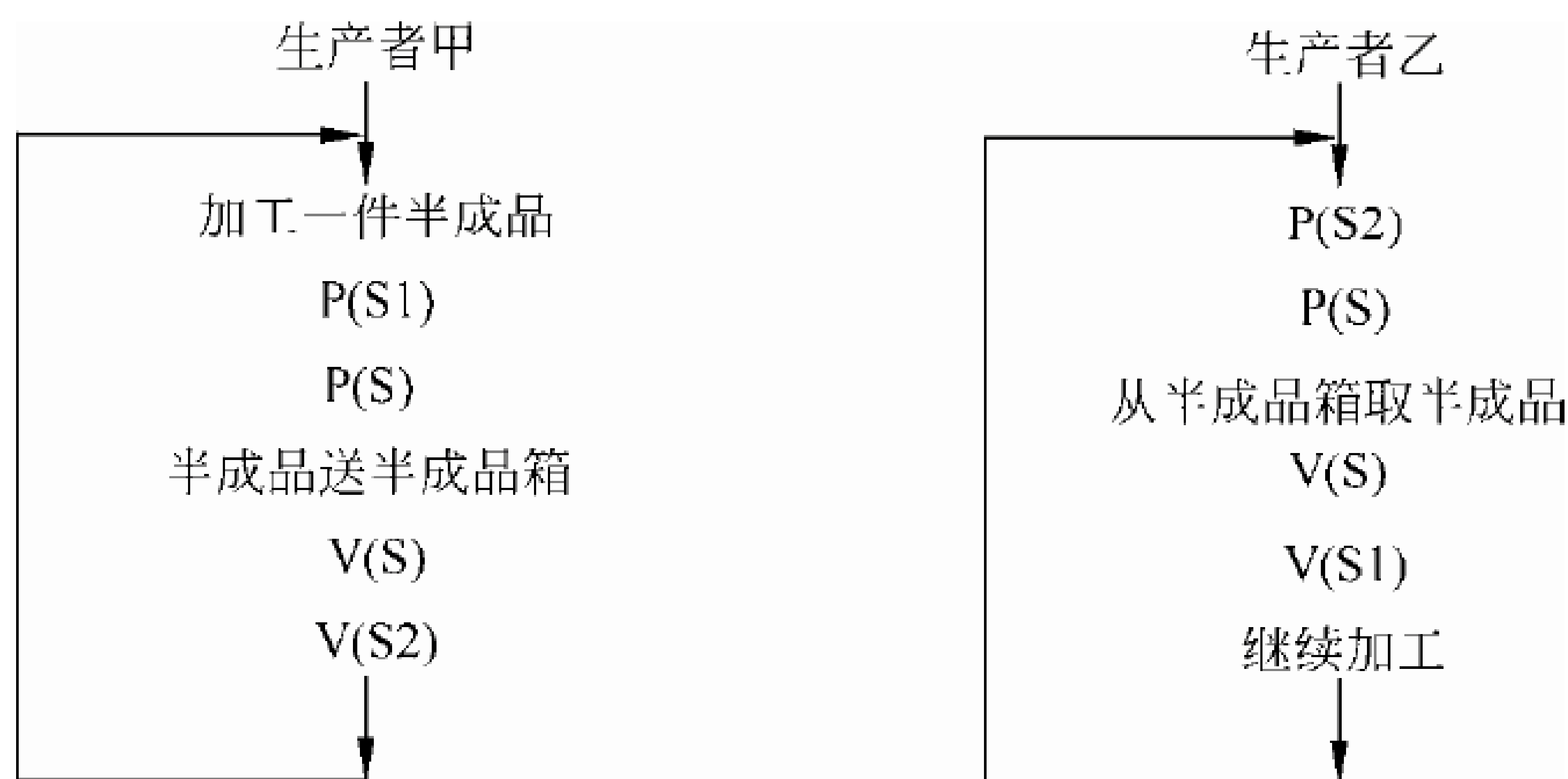
二维数组 arr[1..M, 1..N]的元素可以按行存储，也可以按列存储。按列存储时，元素的排列次序为，先是第一列的所有元素，然后是第二列的所有元素，最后是第 N 列的所有元素。每一列的元素则按行号从小到大依次排列。因此，对于元素 arr[i, j]，其存储位置如下计算：先计算其前面 j–1 列上的元素总数，为(j–1)*M，然后计算第 j 列上排列在 arr[i, j]之前的元素数目，为 i–1，因此 arr[i, j]的地址为 base+((j–1)*M+i–1)*K。

参考答案

（21）C

试题（22）、（23）

某企业生产流水线 M 共有两位生产者，生产者甲不断地将其工序上加工的半成品放入半成品箱，生产者乙从半成品箱取出继续加工。假设半成品箱可存放 *n* 件半成品，采用 PV 操作实现生产者甲和生产者乙的同步可以设置三个信号量 S、S1 和 S2，其同步模型如下图所示。



信号量S是一个互斥信号量，初值为__(22)__；S1、S2的初值分别为__(23)__。

（22）A．0　　B．1　　C．n　　D．任意正整数

（23）A．n、0　　B．0、n　　C．1、n　　D．n、1

试题（22）、（23）分析

由于信号量S是一个互斥信号量，表示半成品箱当前有无生产者使用，所以初值为1。

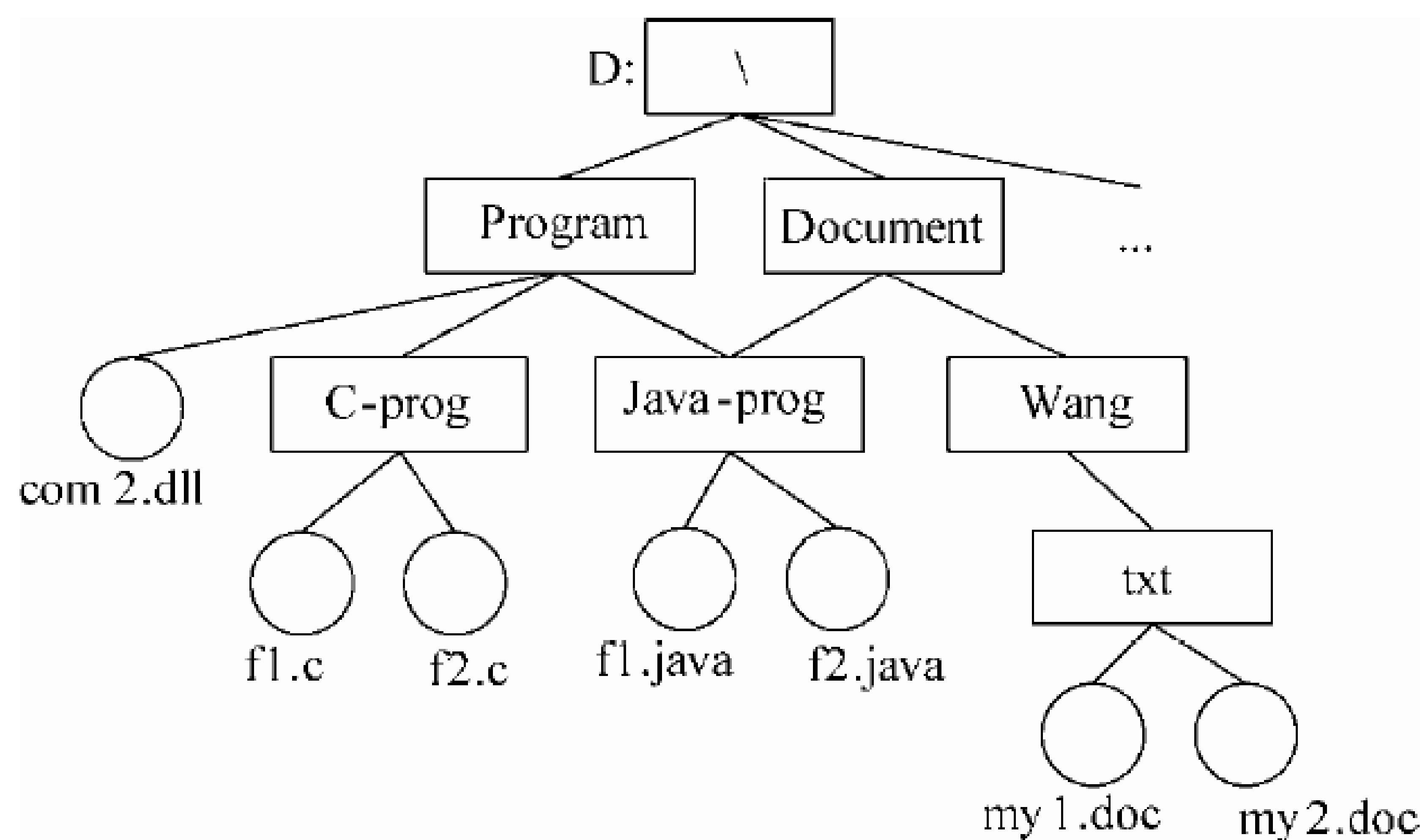
信号量S1表示半成品箱容量，故其初值为*n*。当生产者甲不断地将其工序上加工的半成品放入半成品箱时，应该先测试半成品箱是否有空位，故生产者甲使用P(S1)。信号量S2表示半成品箱有无半成品，初值为0。当生产者乙从半成品箱取出继续加工前应先测试半成品箱有无半成品，故生产者乙使用P(S2)。

参考答案

（22）B　（23）A

试题（24）、（25）

若某文件系统的目录结构如下图所示，假设用户要访问文件f1.java，且当前工作目录为Program，则该文件的全文件名为__(24)__，其相对路径为__(25)__。



（24）A．f1.java　　B．\Document\Java-prog\f1.java

C．D:\Program\Java-prog\f1.java　　D．\Program\Java-prog\f1.java

（25）A．Java-prog\　　B．\Java-prog\

C．Program\Java-prog　　D．\Program\Java-prog\

试题（24）、（25）分析

文件的全文件名应包括盘符及从根目录开始的路径名，所以从题图可以看出文件f1.java的全文件名为D:\Program\Java-prog\f1.java。

文件的相对路径是当前工作目录下的路径名，所以从题图可以看出文件f1.java的相对路径名为Java-prog\。

参考答案

（24）C　（25）A

试题（26）、（27）

假设磁盘每磁道有 18 个扇区，系统刚完成了 10 号柱面的操作，当前移动臂在 13 号柱面上，进程的请求序列如下表所示。若系统采用 SCAN（扫描）调度算法，则系统响应序列为（26）；若系统采用 CSCAN（单向扫描）调度算法，则系统响应序列为（27）。

请求序列	柱面号	磁头号	扇区号
①	15	8	9
②	20	6	5
③	30	9	6
④	20	10	5
⑤	5	4	5
⑥	2	7	4
⑦	15	8	1
⑧	6	3	10
⑨	8	7	9
⑩	15	10	4

（26）A. ⑦⑩①②④③⑨⑧⑤⑥　B. ①⑦⑩②③④⑥⑤⑧⑨
C. ⑦⑩①②④③⑥⑤⑧⑨　D. ①⑦⑩②③④⑧⑨⑥⑤

（27）A. ⑦⑩①②④③⑨⑧⑤⑥　B. ①⑦⑩②③④⑥⑤⑧⑨
C. ⑦⑩①②④③⑥⑤⑧⑨　D. ①⑦⑩②③④⑧⑨⑥⑤

试题（26）、（27）分析

试题（26）分析如下：

当进程请求读磁盘时，操作系统先进行移臂调度，再进行旋转调度。由于系统刚完成了 10 号柱面的操作，当前移动臂在 13 号柱面上，若系统采用 SCAN（扫描）调度算法，则系统响应柱面序列为 15→20→30→8→6→5→2。

按照旋转调度的原则进程在 15 号柱面上的响应序列为⑦→⑩→①，因为进程访问的是不同磁道上的不同编号的扇区，旋转调度总是让首先到达读写磁头位置下的扇区先进行传送操作。进程在 20 号柱面上的响应序列为②→④，或④→②。对于②和④可以任选一个进行读写，因为进程访问的是不同磁道上具有相同编号的扇区，旋转调度可以任选一个读写磁头位置下的扇区进行传送操作。

从上分析可以得出按照 SCAN（扫描）调度算法的响应序列为⑦⑩①②④③⑨⑧⑤⑥。

试题（27）分析如下：

若系统采用 CSCAN（单向扫描）调度算法，在返程时是不响应用户请求的，因此系统的柱面响应序列为 15→20→30→2→5→6→8。

可见，按照 CSCAN（单向扫描）调度算法的响应序列为⑦⑩①②④③⑥⑤⑧⑨。

参考答案

（26）A （27）C

试题（28）

某程序设计语言规定在源程序中的数据都必须具有类型，然而，__（28）__并不是做出此规定的理由。

（28）A．为数据合理分配存储单元
B．可以定义和使用动态数据结构
C．可以规定数据对象的取值范围及能够进行的运算
D．对参与表达式求值的数据对象可以进行合法性检查

试题（28）分析

本题考查程序语言基础知识。

在机器层面上，所有的数据都是二进制形式的。应用领域中的数据可以有不同的形式、意义和运算，程序中的数据已经进行了抽象，不同类型的数据需要不同大小的存储空间，因此为程序中的数据规定类型后，可以更合理地安排存储空间。不同类型的数据其取值方式和运算也不同，引入类型信息后，在对源程序进行编译时就可以对参与表达式求值的数据对象可以进行合法性检查。

参考答案

（28）B

试题（29）

以下关于喷泉模型的叙述中，不正确的是__（29）__。

（29）A．喷泉模型是以对象作为驱动的模型，适合于面向对象的开发方法
B．喷泉模型克服了瀑布模型不支持软件重用和多项开发活动集成的局限性
C．模型中的开发活动常常需要重复多次，在迭代过程中不断地完善软件系统
D．各开发活动（如分析、设计和编码）之间存在明显的边界

试题（29）分析

本题考查软件生存周期模型。

喷泉模型是典型的面向对象生命周期模型，是一种以用户需求为动力，以对象作为驱动的模型。该模型克服了瀑布模型不支持软件重用和多项开发活动集成的局限性。“喷泉”一词本身体现了迭代和无间隙特性。迭代意味着模型中的开发活动常常需要重复多次，在迭代过程中不断地完善软件系统；无间隙是指在开发活动之间不存在明显的边界。

参考答案

（29）D

试题（30）

若全面采用新技术开发一个大学记账系统，以替换原有的系统，则宜选择采用__（30）__

进行开发。

（30）A．瀑布模型　B．演化模型　C．螺旋模型　D．原型模型

试题（30）分析

本题考查软件生存周期模型。

瀑布模型是将软件生存周期各个活动规定为依线性顺序连接的若干阶段的模型，它为软件的开发和维护提供了一种有效的管理模式，适合于软件需求很明确的软件项目的模型。本题中开发的大学记账系统是基于原有系统开发的，要求采用新技术，而需求是明确的。

演化模型在获取一组基本的需求后，通过快速分析构造出该软件的一个初始可运行版本，然后逐步演化成为最终软件产品。原型模型快速构造软件的原型，在此基础上开发最终软件产品。这两类模型主要是针对需求不确定或者不清楚的情况下，进行项目开发建议采用的。而螺旋模型增加了风险分析。

参考答案

（30）A

试题（31）

将每个用户的数据和其他用户的数据隔离开，是考虑了软件的__（31）__质量特性。

（31）A．功能性　B．可靠性　C．可维护性　D．易使用性

试题（31）分析

本题考查需求分析的相关知识。

要求将每个用户的数据和其他用户的数据隔离开，是安全性要求，而安全性质量子特性在 ISO/IEC 软件质量模型中属于功能性质量特性。

参考答案

（31）A

试题（32）

在软件评审中，设计质量是指设计的规格说明书符合用户的要求。设计质量的评审内容不包括__（32）__。

（32）A．软件可靠性　B．软件的可测试性

C．软件性能实现情况　D．模块层次

试题（32）分析

本题考查软件设计的相关知识。

为了使用户满意，软件应该满足两个必要条件：设计的规格说明书符合用户的要求，这称为设计质量；程序按照设计规格说明所规定的情况正确执行，这称为程序质量。

设计质量评审的对象是在需求分析阶段产生的软件需求规格说明、数据需求规格说明、在软件概要设计阶段产生的软件概要设计说明书等。主要从以下方面进行评审：软件的规格说明是否合乎用户的要求；可靠性；保密措施实现情况等；操作特性实施情况

等；性能实现情况；可修改性、可扩充性、可互换性和可移植性；可测试性；可复用性。

参考答案

（32）D

试题（33）

针对应用在运行期的数据特点，修改其排序算法使其更高效，属于__（33）__维护。

（33）A．正确性　　B．适应性　　C．完善性　　D．预防性

试题（33）分析

本题考查软件维护的相关知识。

软件维护的类型一般有四类：正确性维护是指改正在系统开发阶段已发生而系统测试阶段尚未发现的错误；适应性维护是指使应用软件适应信息技术变化和管理需求变化而进行的修改；完善性维护是为扩充功能和改善性能而进行的修改；预防性维护是为了改进应用软件的可靠性和可维护性，为了适应未来变化的软硬件环境的变化，主动增加预防性的新的功能，以适应将来各类变化。

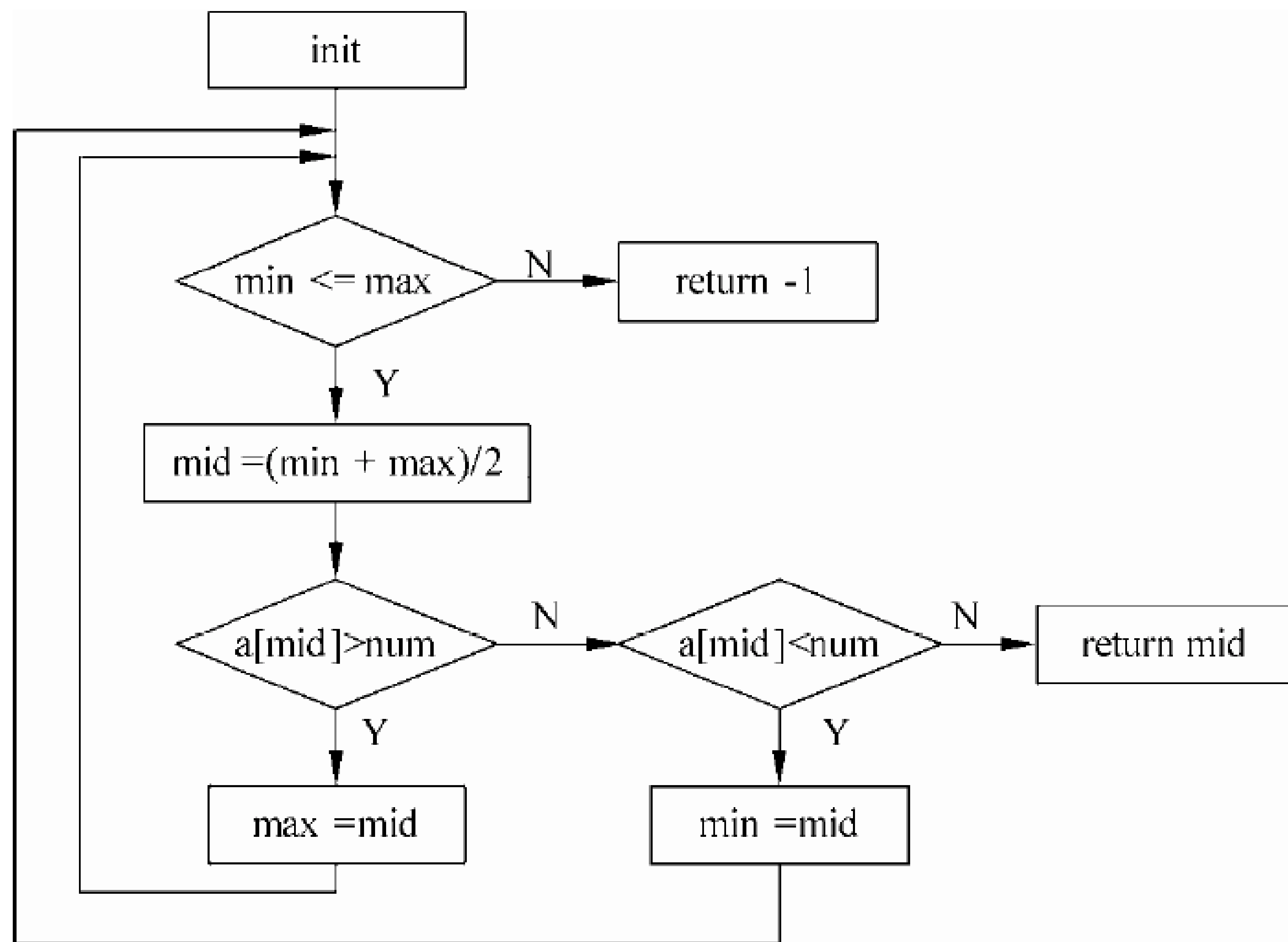
修改现有应用软件中的某个排序算法，提供其运行效率属于完善性维护。

参考答案

（33）C

试题（34）

下图所示的逻辑流实现折半查找功能，最少需要__（34）__个测试用例可以覆盖所有的可能路径。



（34）A．1　　B．2　　C．3　　D．4

试题（34）分析

本题考查软件测试的相关知识。

折半查找是在一组有序的数（假设为递增顺序）中查找一个数的算法，其思想是：将待查找的数与数组中间位置 mid 的数进行比较，若相等，则查找成功；若大于中间位置的数，则在后半部分进行查找；若小于中间位置的数，则在前半部分进行查找。直到查找成功，返回所查找的数的位置，或者失败，返回–1。设计一个查找成功的测试用例，可以覆盖除了 return–1 之外的所有语句和路径；设计一个查找失败的测试用例，可以覆盖除了 return mid 之外的所有语句和路径。因此，最少需要 2 个测试用例才可以覆盖所有的路径。

参考答案

（34）B

试题（35）

在某班级管理系统中，班级的班委有班长、副班长、学习委员和生活委员，且学生年龄在 15～25 岁。若用等价类划分来进行相关测试，则＿（35）＿不是好的测试用例。

（35）A．（队长，15）　　B．（班长，20）

C．（班长，15）　　D．（队长，12）

试题（35）分析

本题考查软件测试的相关知识。

等价类划分是一类黑盒测试技术，将程序的输入域划分为若干等价类，然后从每个等价类中选取一个代表性数据作为测试用例。本题的等价类划分可以划分为三个等价类，一个有效等价类 I，即班委来自集合{班长，副班长，学习委员，生活委员}，年龄在 15～25；一个无效等价类 II，即班委不来自集合{班长，副班长，学习委员，生活委员}，而年龄在 15～25；一个无效等价类III，即班委来自集合{班长，副班长，学习委员，生活委员}，而年龄不在 15～25。题中选项 A 来自等价类 II，选项 B 和选项 C 来自等价类 I，而选项 D 则不属于任何等价类，因此不是一个好的测试用例。

参考答案

（35）D

试题（36）

进行防错性程序设计，可以有效地控制＿（36）＿维护成本。

（36）A．正确性　　B．适应性　　C．完善性　　D．预防性

试题（36）分析

本题考查软件维护的相关知识。

软件维护的类型一般有四类：正确性维护、适应性维护、完善性维护和预防性维护。防错性的程序设计可以减少在系统运行时发生错误，因此可以有效地控制正确性维护成本。

参考答案

（36）A

试题（37）

采用面向对象开发方法时，对象是系统运行时基本实体。以下关于对象的叙述中，正确的是__（37）__。

（37）A．对象只能包括数据（属性）
B．对象只能包括操作（行为）
C．对象一定有相同的属性和行为
D．对象通常由对象名、属性和操作三个部分组成

试题（37）分析

本题考查面向对象的基本知识。

采用面向对象开发方法时，对象是系统运行时基本实体。它既包括数据（属性），也包括作用于数据的操作（行为）。一个对象通常可由对象名、属性和操作三部分组成。

参考答案

（37）D

试题（38）、（39）

一个类是__（38）__。在定义类时，将属性声明为 private 的目的是__（39）__。

（38）A．一组对象的封装 B．表示一组对象的层次关系
C．一组对象的实例 D．一组对象的抽象定义

（39）A．实现数据隐藏，以免意外更改
B．操作符重载
C．实现属性值不可更改
D．实现属性值对类的所有对象共享

试题（38）、（39）分析

本题考查面向对象的基本知识。

面向对象技术中，将一组大体上相似的对象定义为一个类。把一组对象的共同特征加以抽象并存储在一个类中，是面向对象技术中的一个重要特点。一个所包含的方法和数据描述一组对象的共同行为和属性。在类定义时，属性声明 private 的目的是实现数据隐藏，以免意外更改。

参考答案

（38）D （39）A

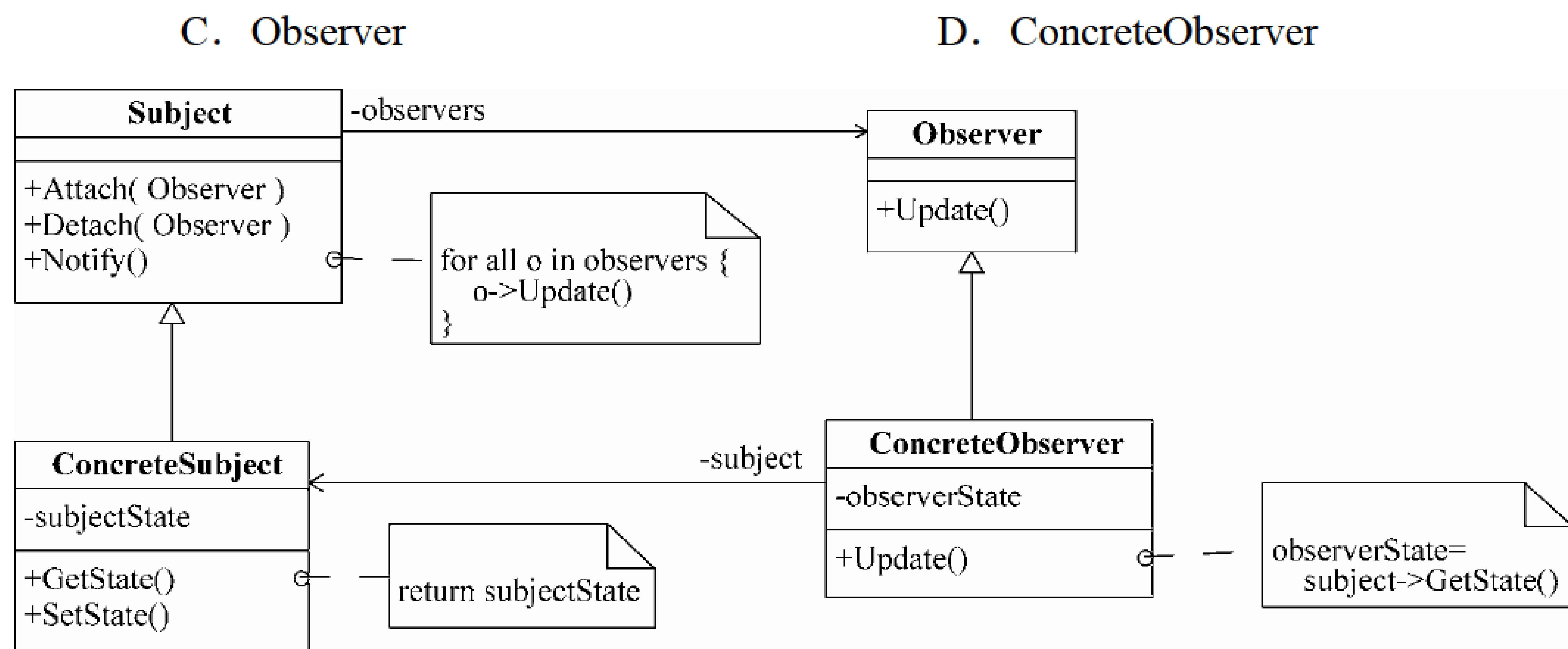
试题（40）、（41）

__（40）__设计模式允许一个对象在其状态改变时，通知依赖它的所有对象。该设计模式的类图如下图，其中，__（41）__在其状态发生改变时，向它的各个观察者发出通知。

（40）A．命令（Command） B．责任链（Chain of Responsibility）
C．观察者（Observer） D．迭代器（Iterator）

（41）A．Subject B．ConcreteSubject

C．Observer　　　　　　　　　D．ConcreteObserver



试题（40）、（41）分析

本题考查设计模式的基本知识。

命令（Command）模式通过将请求封装为一个对象，可将不同的请求对客户进行参数化。责任链（Chain of Responsibility）模式将多个对象的请求连成一条链，并沿着这条链传递该请求，直到有一个对象处理它为止，避免请求的发送者和接收者之间的耦合关系。观察者（Observer）模式定义对象之间的一种一对多的依赖关系，当一个对象的状态发生改变时，所有依赖于它的对象都得到通知并被自动更新。

在上述观察者模式的类图中，Subject（目标）知道其观察者，可以有任意多个观察者观察同一个目标，提供注册和删除观察者对象的接口。Observer（观察者）为那些在目标发生改变时需获得通知的对象定义一个更新接口。ConcreteSubject（具体目标）将有关状态存入各 ConcreteObserver 对象，当它的状态发生改变时，向它的各个观察者发出通知。ConcreteObserver（具体观察者）维护一个指向 ConcreteSubject 对象的引用，存储有关状态，实现 Observer 的更新接口以使自身状态与目标的状态保持一致。

参考答案

（40）C　（41）B

试题（42）

在面向对象软件开发中，封装是一种__（42）__技术，其目的是使对象的使用者和生产者分离。

（42）A．接口管理　　B．信息隐藏　　C．多态　　　D．聚合

试题（42）分析

本题考查面向对象的基础知识。

在面向对象软件开发中，对象是软件系统中基本的运行时实体，对象封装了属性和行为。封装是一种信息隐藏技术，其目的是使对象的使用者和生产者分离，使对象的定义和实现分开。

参考答案

（42）B

试题（43）

欲动态地给一个对象添加职责，宜采用__（43）__模式。

（43）A．适配器（Adapter）　　B．桥接（Bridge）
　　C．组合（Composite）　　D．装饰器（Decorator）

试题（43）分析

本题考查设计模式的基本知识。

适配器（Adapter）模式是将类的接口转换成客户希望的另外一个接口，使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。Bridge（桥接）模式将对象的抽象和其实现分离，从而可以独立地改变它们。组合（Composite）模式描述了如何构造一个类层次式结构。装饰器（Decorator）模式的意图是动态地给一个对象添加一些额外职责。在需要给某个对象而不是整个类添加一些功能时使用。这种模式对增加功能比生成子类更加灵活。

参考答案

（43）D

试题（44）

__（44）__模式通过提供与对象相同的接口来控制对这个对象的访问。

（44）A．适配器（Adapter）　　B．代理（Proxy）
　　C．组合（Composite）　　D．装饰器（Decorator）

试题（44）分析

本题考查设计模式的基本知识。

适配器（Adapter）模式是将类的接口转换成客户希望的另外一个接口。代理（Proxy）模式通过提供与对象相同的接口来控制对这个对象的访问，以使得在确实需要这个对象时才对它进行创建和初始化。组合（Composite）模式描述了如何构造一个类层次式结构。装饰器（Decorator）模式动态地给一个对象添加职责。

参考答案

（44）B

试题（45）

采用 UML 进行面向对象开发时，部署图通常在__（45）__阶段使用。

（45）A．需求分析　　B．架构设计　　C．实现　　D．实施

试题（45）分析

本题考查 UML 面向对象开发的基本知识。

UML 2.0 提供多种视图，只有部署图描述系统的物理视图。部署图通常在实施阶段使用，以说明哪些组件或子系统部署于哪些结点。

参考答案

（45）D

试题（46）、（47）

业务用例和参与者一起描述__（46）__，而业务对象模型描述__（47）__。

（46）A．工作过程中的静态元素　　B．工作过程中的动态元素
C．工作过程中的逻辑视图　　D．组织支持的业务过程

（47）A．业务结构
B．结构元素如何完成业务用例
C．业务结构以及结构元素如何完成业务用例
D．组织支持的业务过程

试题（46）、（47）分析

本题考查面向对象建模的基本知识。

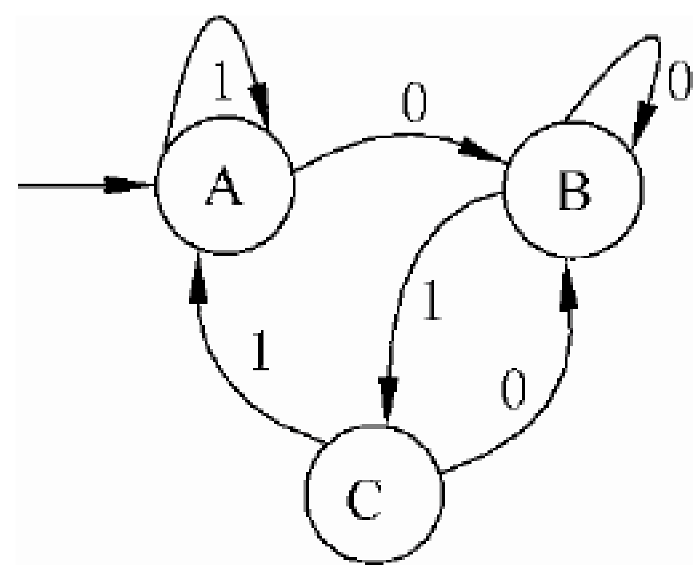
在采用面向对象方法进行业务建模时，业务用例和参与者一起描述组织或企业所支持的业务过程。业务流程被定义为多个不同的业务用例，其中每个业务用例都代表业务中某个特定的工作流程。业务用例确定了执行业务时将要发生的事情，描述了一系列动作的执行，以及产生对特定业务主角具有价值的结果。业务对象模型从业务角色内部的观点定义了业务用例。该模型确定了业务人员及其处理和使用的对象之间应该具有的静态和动态关系，注重业务中承担的角色及其当前职责，既描述业务结构，又描述这些结构元素如何完成业务用例。

参考答案

（46）D　（47）C

试题（48）

下图所示为一个有限自动机（其中，A 是初态、C 是终态），该自动机识别的语言可用正规式__（48）__表示。



（48）A．(0|1)*01　　B．1*0*10*1　　C．1*(0)*01　　D．1*(0|10)*1*

试题（48）分析

本题考查程序语言翻译基础知识。

分析题中所给自动机识别字符串的特点可知，该自动机识别的字符串必须以 01 结尾，而之前的 0 和 1 可以以任意方式组合，因此，正规式为(0|1)*01。

参考答案

（48）A

试题（49）、（50）

函数 t、f 的定义如下所示，其中，a 是整型全局变量。设调用函数 t 前 a 的值为 5，则在函数 t 中以传值调用（call by value）方式调用函数 f 时，输出为__（49）__；在函数 t 中以引用调用（call by reference）方式调用函数 f 时，输出为__（50）__。

```
t( ):  | int x = f(a);  |        f(int r):  | a = r+1;   r = r * 2 ; |
       | print a+x;     |                   | return r;               |
```

（49）A．12　　B．16　　C．20　　D．24

（50）A．12　　B．16　　C．20　　D．24

试题（49）、（50）分析

本题考查函数调用时参数传递基础知识。

发生函数调用时，调用函数与被调用函数之间交换信息的主要方法有传值调用和引用调用两种。

若实现函数调用时实参向形式参数传递相应类型的值，则称为是传值调用。这种方式下形式参数不能向实际参数传递信息。

在 C 语言中，要实现被调用函数对实际参数的修改，必须用指针作形参。即调用时需要先对实参进行取地址运算，然后将实参的地址传递给指针形参。本质上仍属于传值调用。

引用是 C++中增加的数据类型，当形式参数为引用类型时，函数中对形参的访问和修改本质上就是针对相应实际参数变量所作的访问和改变。

本题中，传值调用方式下，表达式“x = f(a)”中调用 f 时，是将 a 的值（即 5）传给 r，这样执行函数 f 时，r 的初始值为 5，经过“a = r+1”运算后，全局变量 a 的值从 5 变为 6，然后“r = r*2”将 r 的值改变为 10，“return r”将 10 返回并赋值给 x，因此执行“print a+x”后输出了 16。

传值调用方式下，表达式“x = f(a)”中调用 f 时，r 则是 a 的引用（即 r 是 a 的别名），因此，经过“a = r+1”运算后，a 的值（也就是 r 的值）变为 6，然后“r = r*2”将 r 的值（也就是 a 的值）改变为 12，“return r”使得将 12 返回并赋值给 x，因此执行“print a+x”后输出了 24。

参考答案

（49）B　（50）D

试题（51）、（52）

将 Students 表的插入权限赋予用户 UserA，并允许其将该权限授予他人，应使用的 SQL 语句为：

```
GRANT____(51)____TABLE Students TO UserA____(52)____;
```

（51）A．UPDATE　　B．UPDATE ON　　C．INSERT　　D．INSERT ON

（52）A．FOR ALL　　B．PUBLIC

C．WITH CHECK OPTION　　D．WITH GRANT OPTION

试题（51）、（52）分析

本题考查关系代数运算与 SQL 查询方面的基础知识。

授权语句的格式如下：

```
GRANT <权限>[,<权限>]…[ON<对象类型><对象名>]
   TO <用户>[,<用户]>]… [WITH GRANT OPTION];
```

若在授权语句中指定了“WITH GRANT OPTION”子句，那么，获得了权限的用户还可以将该权限赋给其他用户。

参考答案

（51）D　（52）D

试题（53）～（55）

若有关系 R（A，B，C，D）和 S（C，D，E），则与表达式 $\pi_{3,4,7}(\sigma_{4<5}(R\times S))$ 等价的 SQL 语句如下：

```
SELECT____(53)____FROM____(54)____WHERE____(55)____;
```

（53）A．A,B,C,D,E　　B．C,D,E

C．R.A, R.B,R.C, R.D, S.E　　D．R.C, R.D,S.E

（54）A．R　　B．S　　C．R, S　　D．RS

（55）A．D < C　　B．R.D < S.C　　C．R.D< R.C　　D．S.D< R.C

试题（53）～（55）分析

本题考查关系代数运算与 SQL 查询方面的基础知识。

在 $\pi_{3,4,7}(\sigma_{4<5}(R\times S))$ 中，R×S 的属性列名分别为：R.A、R.B、R.C、R.D、S.C、S.D、和 S.E，$\pi_{3,4,7}(\sigma_{4<5}(R\times S))$ 的含义是从 R×S 结果集中选取 R.D<S.C 的元组，再进行 R.C、R.D 和 S.E 投影。

参考答案

（53）D　（54）C　（55）B

试题（56）

E-R 图转换为关系模型时，对于实体 E_1 与 E_2 间的多对多联系，应该将____(56)____。

（56）A．E_1 的码加上联系上的属性并入 E_2

B．E_1 的码加上联系上的属性独立构成一个关系模式

C．E_2 的码加上联系上的属性独立构成一个关系模式

D. E_1 与 E_2 码加上联系上的属性独立构成一个关系模式

试题（56）分析

本题考查 E-R 图向关系模型转换方面的基础知识。

E-R 图转换为关系模型时，对两个实体间的多对多联系应该转换为一个独立的关系模式，其方法是将两个实体的码加上联系的属性构成一个独立的关系模式。

参考答案

（56）D

试题（57）

在 KMP 模式匹配算法中，需要求解模式串 p 的 next 函数值，其定义如下（其中，j 是字符在模式串中的序号）。对于模式串“abaabaca”，其 next 函数值序列为 __（57）__ 。

$$next[j]=\begin{cases}0 & j=1\\ \max\{k \mid 1<k<j,'p_1p_2\cdots p_{k-1}'='p_{j-k+1}p_{j-k+2}\cdots p_{j-1}'\} & \\ 1 & 其他情况\end{cases}$$

（57）A. 01111111　　B. 01122341　　C. 01234567　　D. 01122334

试题（57）分析

本题考查字符串基本运算。

根据 next 函数的定义，“abaabaca”的 next 函数值为“01122341”。

参考答案

（57）B

试题（58）

对于线性表（由 n 个同类元素构成的线性序列），采用单向循环链表存储的特点之一是 __（58）__ 。

（58）A. 从表中任意结点出发都能遍历整个链表

B. 对表中的任意结点可以进行随机访问

C. 对于表中的任意一个结点，访问其直接前驱和直接后继结点所用时间相同

D. 第一个结点必须是头结点

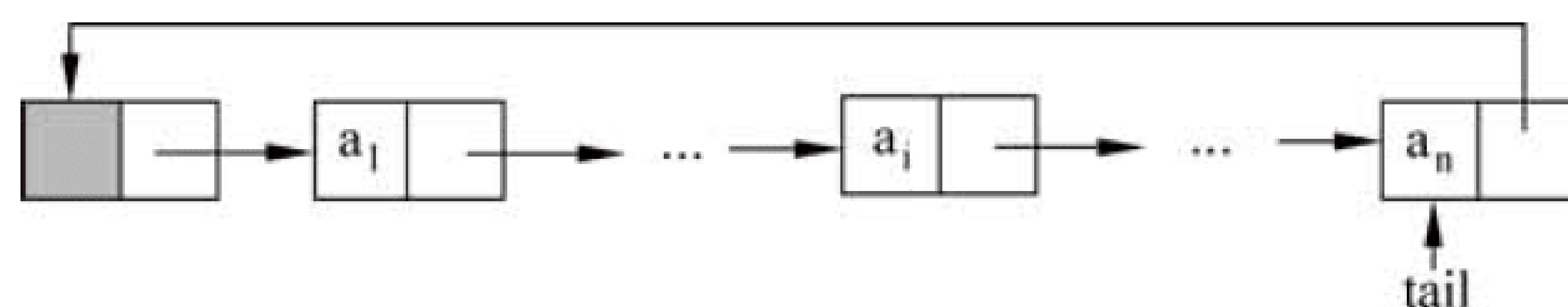
试题（58）分析

本题考查线性表的链表存储结构知识。

随机访问是指可由元素的序号和第一个元素存储位置的首地址计算得出该序号所对应元素的存储位置，这要求这一组元素必须连续地存储，链表存储结构中元素的存储位置是可以分散的，仅通过指针将逻辑上相邻而存储位置不要求相邻的元素链接起来，而且只能顺着指针所指示的方向进行遍历。

单向循环链表中指针的指示方向是单方向地，其示意图如下所示，对于表中的任意一个元素，访问其直接后继的运算时间复杂度为 O(1)，访问其直接前驱的运算时间复杂度为 O(n)。链表中是否含有头结点要看具体的应用情况和运算要求，并没有必须设置的

要求。



参考答案

（58）A

试题（59）

无向图中一个顶点的度是指图中与该顶点相邻接的顶点数。若无向图 G 中的顶点数为 n，边数为 e，则所有顶点的度数之和为__（59）__。

（59）A．n*e　　B．n+e　　C．2n　　D．2e

试题（59）分析

本题考查图结构的基础知识。

对于无向图中的两个顶点 u 和 v，若存在边（u，v），则该边为计算 u 的度和 v 的度各贡献一个值 1，因此，所有顶点的度数之和为 e 的两倍。

参考答案

（59）D

试题（60）

一棵满二叉树，其每一层结点个数都达到最大值，对其中的结点从 1 开始顺序编号，即根结点编号为 1，其左、右孩子结点编号分别为 2 和 3，再下一层从左到右的编号为 4、5、6、7，依此类推，每一层都从左到右依次编号，直到最后的叶子结点层为止，则用__（60）__可判定编号为 m 和 n 的两个结点是否在同一层。

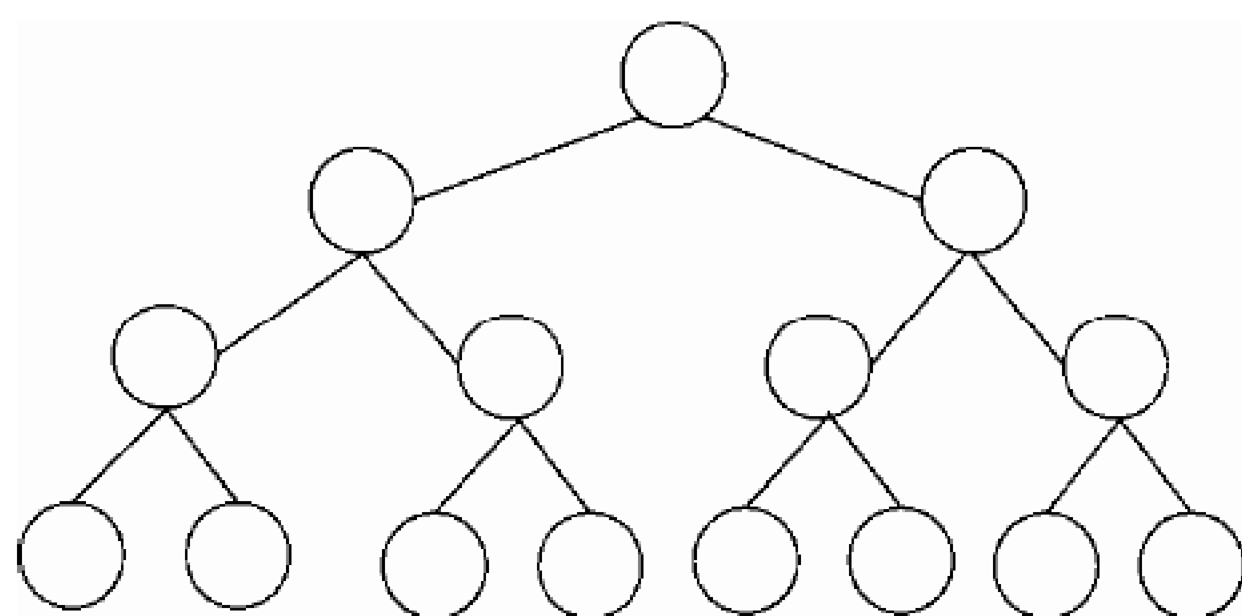
（60）A．$\log_2 m = \log_2 n$　　B．$\lfloor \log_2 m \rfloor = \lfloor \log_2 n \rfloor$

C．$\lfloor \log_2 m \rfloor + 1 = \lfloor \log_2 n \rfloor$　　D．$\lfloor \log_2 m \rfloor = \lfloor \log_2 n \rfloor + 1$

试题（60）分析

本题考查二叉树基础知识。

高度为 h 的满二叉树的结点个数为 $2^h - 1$，高度为 4 的满二叉树如下图所示。



从该例可知，第一层结点的编号为 $2^1 - 1$，第二层的结点编号为 $2^1 \sim 2^2 - 1$，第三层的结点编号为 $2^2 \sim 2^3 - 1$，…，第 i 层的结点编号为 $2^{i-1} \sim 2^i - 1$。因此，对于编号为 m 的

结点，其所在层次为$\lfloor \log_2 m \rfloor+1$，对于编号为 n 的结点，其所在层次为$\lfloor \log_2 n \rfloor+1$，所以用$\lfloor \log_2 m \rfloor = \lfloor \log_2 n \rfloor$可以判断这两个结点是否在同一层。

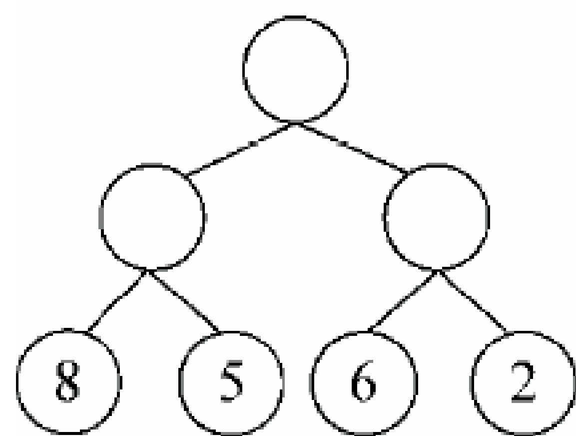
参考答案

（60）B

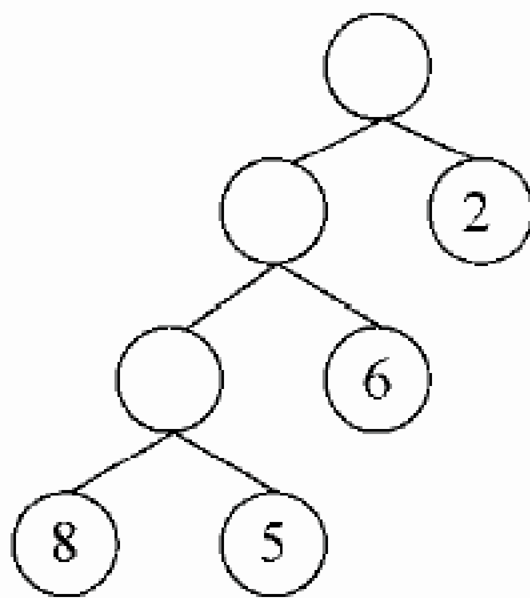
试题（61）

__（61）__是由权值集合{8，5，6，2}构造的哈夫曼树（最优二叉树）。

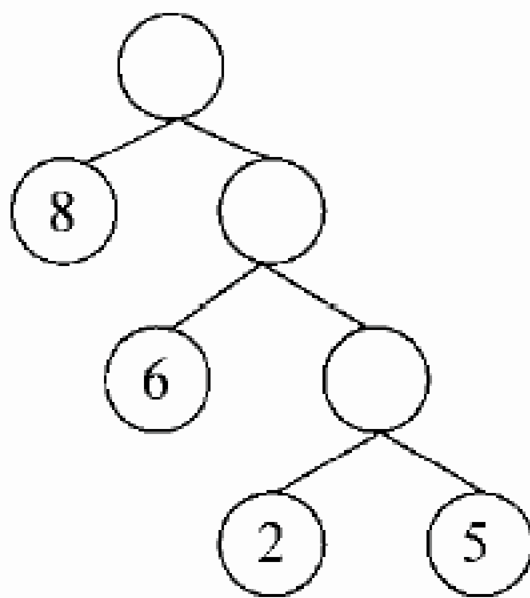
（61）A.



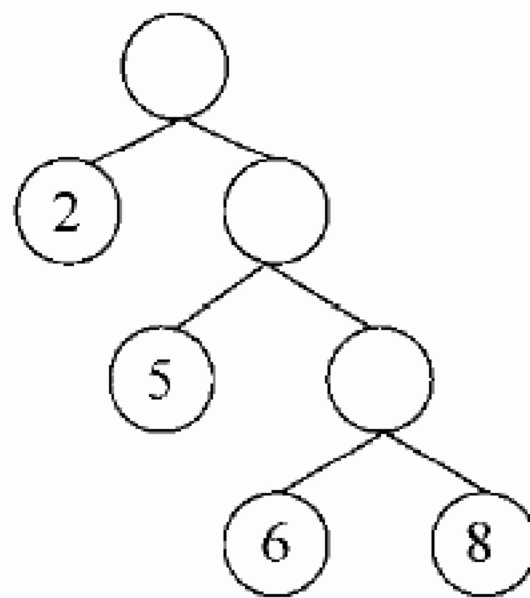
B.



C.



D.



试题（61）分析

本题考查二叉树应用知识。

构造最优二叉树的哈夫曼算法如下：

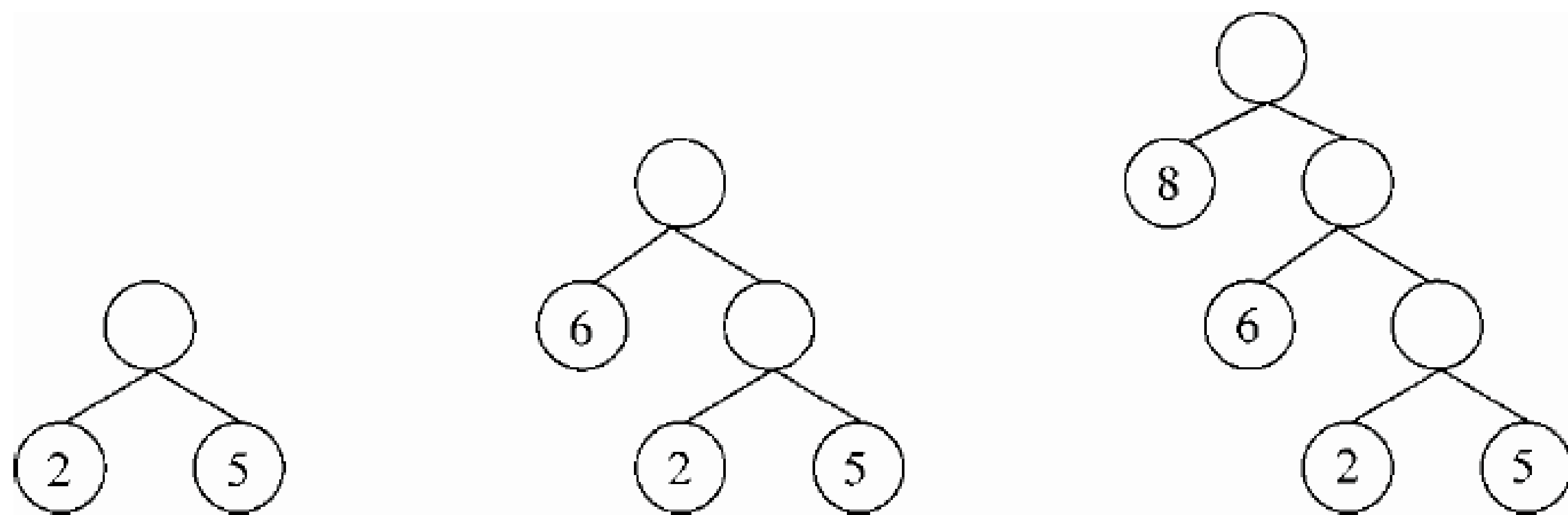
① 根据给定的 n 个权值{w_1，w_2，…， w_n}，构成 n 棵二叉树的集合 F={T_1，T_2，…，T_n}，其中每棵二叉树 T_i 中只有一个带权为 w_i 的根结点，其左右子树均空。

② 在 F 中选取两棵权值最小的二叉树作为左、右子树构造一棵新的二叉树，置新构造二叉树的根结点的权值为其左、右子树根结点的权值之和。

③ 从 F 中删除这两棵树，同时将新得到的二叉树加入到 F 中。

重复②、③，直到 F 中只含一棵树时为止。这棵树便是最优二叉树（哈夫曼树）。

根据题中给出的权值集合，构造哈夫曼树的过程如下图所示。



参考答案

（61）C

试题（62）

迪杰斯特拉（Dijkstra）算法用于求解图上的单源点最短路径。该算法按路径长度递增次序产生最短路径，本质上说，该算法是一种基于__（62）__策略的算法。

（62）A．分治 B．动态规划 C．贪心 D．回溯

试题（62）分析

本题考查算法的设计策略。

单源点最短路径问题是指给定图 G 和源点 v_0，求从 v_0 到图 G 中其余各顶点的最短路径。迪杰斯特拉（Dijkstra）算法是一个求解单源点最短路径的经典算法，其思想是：把图中所有的顶点分成两个集合 S 和 T，S 集合开始时只包含顶点 v_0，T 集合开始时包含图中除了顶点 v_0 之外的所有顶点。凡是以 v_0 为源点，已经确定了最短路径的终点并入 S 集合中，顶点集合 T 则是尚未确定最短路径的顶点集合。按各顶点与 v_0 间最短路径长度递增的次序，逐个把 T 集合中的顶点加入到 S 集合中，使得从 v_0 到 S 集合中各顶点的路径长度始终不大于从 v_0 到 T 集合中各顶点的路径长度。该算法是以一种贪心的方式将 T 集合中的顶点加入到 S 集合中的，而且该贪心方法可以求得问题的最优解。

参考答案

（62）C

试题（63）

在有 n 个无序无重复元素值的数组中查找第 i 小的数的算法描述如下：任意取一个元素 r，用划分操作确定其在数组中的位置，假设元素 r 为第 k 小的数。若 i 等于 k，则返回该元素值；若 i 小于 k，则在划分的前半部分递归进行划分操作找第 i 小的数；否则在划分的后半部分递归进行划分操作找第 k-i 小的数。该算法是一种基于<u>（63）</u>策略的算法。

（63）A．分治　　B．动态规划　　C．贪心　　D．回溯

试题（63）分析

本题考查算法的设计策略。

从题干可以看出，划分操作与快速排序中的划分操作是一样的，确定某个元素如 r 的最终位置，划分后，在 r 之前的元素都小于 r，在 r 之后的元素都大于 r（假设无重复元素）。因此可以据此确定 r 是数组中第几小的数。题干所述的算法把找第 i 小的数转换为确定任意一个元素是第几小的数，然后根据这个结果再在依据该元素划分后得到的结果在前一部分还是后一部分来继续确定某个元素为第几小的数，重复这种处理，直到找到第 i 小的数。这是分治策略的一个典型应用。

参考答案

（63）A

试题（64）

对 n 个元素值分别为–1、0 或 1 的整型数组 A 进行升序排序的算法描述如下：统计 A 中–1、0 和 1 的个数，设分别为 n_1、n_2 和 n_3，然后将 A 中的前 n_1 个元素赋值为–1，第 n_1+1 到 n_1+n_2 个元素赋值为 0，最后 n_3 个元素赋值为 1。该算法的时间复杂度和空间复杂度分别为<u>（64）</u>。

（64）A．$\Theta(n)$ 和 $\Theta(1)$　　B．$\Theta(n)$ 和 $\Theta(n)$

C. $\Theta(n^2)$ 和 $\Theta(1)$　　D. $\Theta(n^2)$ 和 $\Theta(n)$

试题（64）分析

本题考查算法的分析技术。

算法首先遍历数组 A 中的所有元素，统计其中–1、0 和 1 的个数，其时间复杂度为 $\Theta(n)$，需要三个额外存储空间，因此空间复杂度 $\Theta(1)$。然后根据前面的统计结果对 A 数组的元素重新赋值，其时间复杂度为 $\Theta(n)$。因此算法的时间复杂度和空间复杂度分别为 $\Theta(n)$ 和 $\Theta(1)$。

参考答案

（64）A

试题（65）

设算法 A 的时间复杂度可用递归式 $T(n)=\begin{cases}\Theta(1) & ,n=1\\ 7T(n/2)+n^2 & ,n>1\end{cases}$ 表示，算法 B 的时间复杂度可用递归式 $T(n)=\begin{cases}\Theta(1) & ,n=1\\ aT(n/4)+n^2 & ,n>1\end{cases}$ 表示，若要使得算法 B 渐进地快于算法 A，则 a 的最大整数为__（65）__。

（65）A. 48　　B. 49　　C. 13　　D. 14

试题（65）分析

本题考查算法的分析技术。

根据主定理，算法 A 的时间复杂度分析如下：a=7，b=2，$\log_b a=\log_2 7>2$，因此属于情况（1），时间复杂度为 $n^{\log_2 7}$。算法 B 的时间复杂度分析：a，b=4，$\log_b a=\log_4 a$，要使算法 B 快于算法 A，则需要 $\log_4 a<\log_2 7$，而 $\log_2 7=\log_4 49$，因此有 $\log_4 a<\log_4 49$，该式成立的最大 a 为 48，因此该题选 A。

参考答案

（65）A

试题（66）、（67）

A 类网络是很大的网络，每个 A 类网络中可以有__（66）__个网络地址。实际使用中必须把 A 类网络划分为子网，如果指定的子网掩码为 255.255.192.0，则该网络被划分为__（67）__个子网。

（66）A. 2^{10}　　B. 2^{12}　　C. 2^{20}　　D. 2^{24}

（67）A. 128　　B. 256　　C. 1024　　D. 2048

试题（66）、（67）分析

A 类网络的地址掩码是 8 比特，剩余的 24 比特可表示主机地址，所以主机地址数为 2^{24} 个。如果为 A 类网络指定的子网掩码为 255.255.192.0，则其二进制表示为 11111111 11111111 11000000 00000000，实际上把 A 类网络划分为 2^{10}=1024 个子网。

参考答案

（66）D　（67）C

试题（68）、（69）

TCP 是互联网中的__（68）__协议，使用__（69）__次握手协议建立连接。

（68）A．传输层　B．网络层　C．会话层　D．应用层

（69）A．1　B．2　C．3　D．4

试题（68）、（69）分析

TCP 是互联网中的传输层协议，使用 3 次握手协议建立连接。这种建立连接的方法可以防止产生错误的连接，这种错误往往是由网络中存储的过期的分组引起的。TCP 使用的流量控制协议是可变大小的滑动窗口协议。

参考答案

（68）A　（69）C

试题（70）

在 Windows 系统中，为排除 DNS 域名解析故障，需要刷新 DNS 解析器缓存，应使用的命令是__（70）__。

（70）A．ipconfig/renew　B．ipconfig/flushdns

C．netstat -r　D．arp -a

试题（70）分析

本题考查 Web 站点文档及相关知识。刷新和重置缓存的命令是 ipconfig /flushdns。

参考答案

（70）B

试题（71）～（75）

Extreme Programming (XP) is a discipline of software development with __（71）__ of simplicity, communication, feedback and courage. Successful software development is a team effort – not just the development team, but the larger team consisting of customer, management and developers. XP is a simple process that brings these people together and helps them to succeed together. XP is aimed primarily at object-oriented projects using teams of a dozen or fewer programmers in one location. The principles of XP apply to any __（72）__ project that needs to deliver quality software rapidly and flexibly.

An XP project needs a(an) __（73）__ customer to provide guidance. Customers, programmers, managers, are all working __（74）__ to build the system that's needed. Customers – those who have software that needs to be developed – will learn simple, effective ways to __（75）__ what they need, to be sure that they are getting what they need, and to steer the project to success.

（71）A．importance　B．keys　C．roles　D．values

（72）A．small-sized　　B．moderately-sized
　　C．large-sized　　D．huge-sized

（73）A．part-time　　B．casual　　C．seldom　　D．full-time

（74）A．together　　B．by themselves　　C．separately　　D．alone

（75）A．tell　　B．know　　C．communicate　　D．feedback

参考译文

极限编程（XP）是一种软件开发方法，其核心价值观是简单、沟通、反馈和勇气。成功的软件开发是团队努力的结果——不仅仅指开发团队，而是包括了客户、管理人员和开发人员组成的更大团队。XP 是一种将上述人员组织起来并帮助他们取得成功的简单的过程。XP 主要针对一个十几人或更少程序员组成的、在同一个场所工作的面向对象的项目团队。XP 原则适用于需要快速且灵活地交付高质量软件的中等规模项目组。

一个 XP 项目组需要一个全程参与的客户给予指导。客户、程序员和项目经理协同工作来构建需要的软件系统。客户，也就是需要软件的人，将学到简单而有效的沟通方法，来确保获得他们所需要的，从而引导项目走向成功。

试题（71）～（75）分析

题目中描述敏捷开发方法极限编程（XP）。XP 强调简单、沟通、反馈和勇气 4 个核心价值（values），适合于需要快速和灵活交付的适当规模（moderately-sized）的任何项目。XP 强调客户全职（full-time）参与。客户和项目的其他成员工作在一起（together），以简单方式进行有效地沟通（communicate），以掌握项目按照需求向项目成功的方向进行。

参考答案

（71）D　（72）B　（73）D　（74）A　（75）C

第 12 章　2011 下半年软件设计师下午试题分析与解答

试题一（共 15 分）

阅读下列说明和图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

【说明】

某公司欲开发招聘系统以提高招聘效率，其主要功能如下：

（1）接受申请

验证应聘者所提供的自身信息是否完整，是否说明了应聘职位，受理验证合格的申请，给应聘者发送致谢信息。

（2）评估应聘者

根据部门经理设置的职位要求，审查已经受理的申请；对未被录用的应聘者进行谢绝处理，将未被录用的应聘者信息存入未录用的应聘者表，并给其发送谢绝决策；对录用的应聘者进行职位安排评价，将评价结果存入评价结果表，并给其发送录用决策，发送录用职位和录用者信息给工资系统。

现采用结构化方法对招聘系统进行分析与设计，获得如图 1-1 所示的顶层数据流图、图 1-2 所示 0 层数据流图和图 1-3 所示 1 层数据流图。

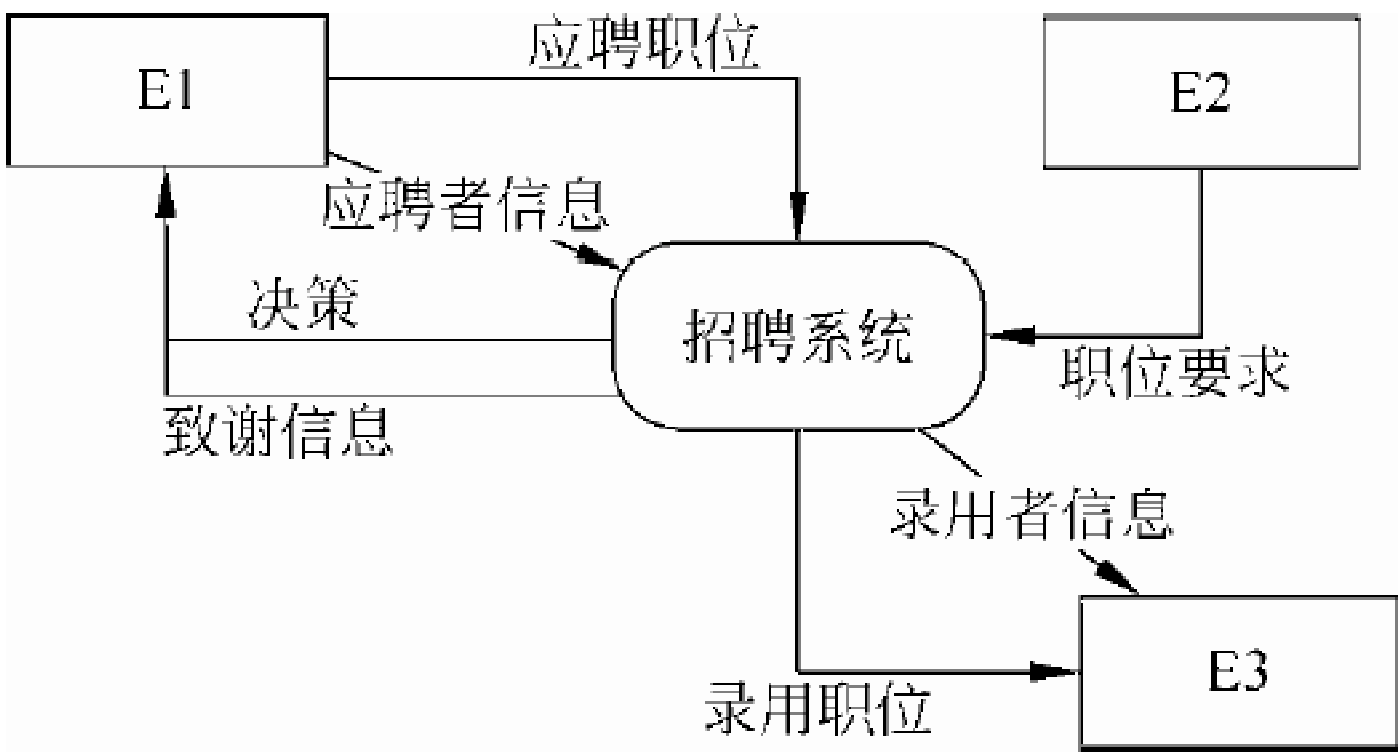


图 1-1　顶层数据流图

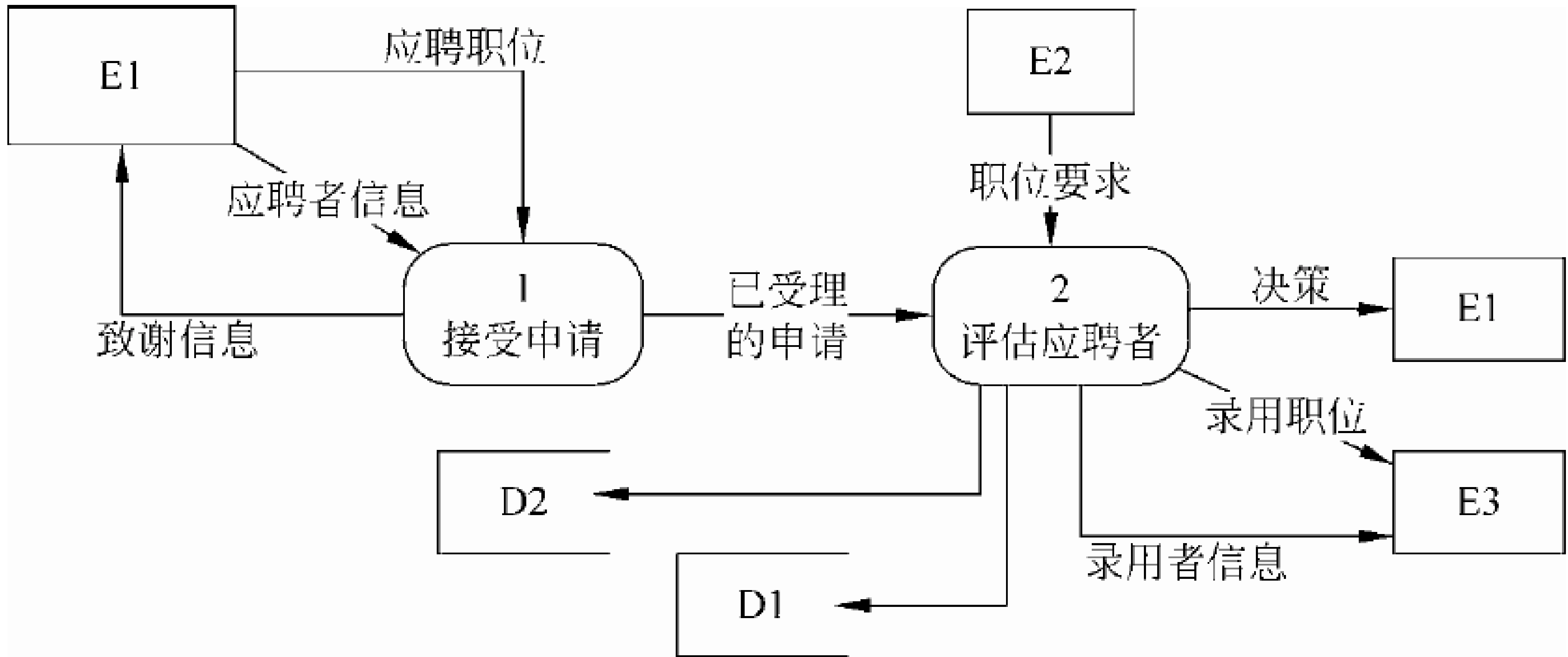


图 1-2　0 层数据流图

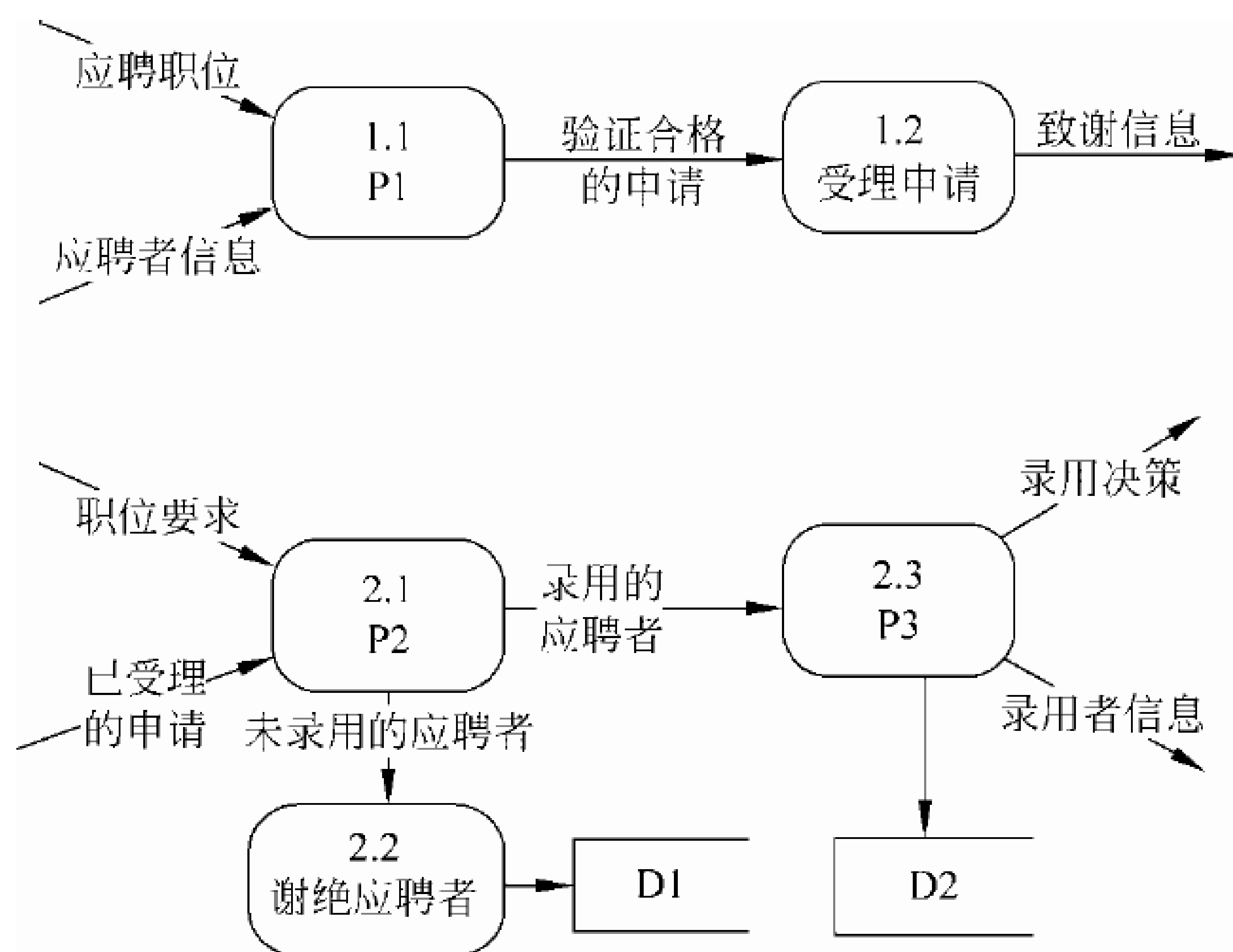


图 1-3　1 层数据流图

【问题 1】（3 分）

使用说明中的术语，给出图中 E1～E3 所对应的实体名称。

【问题 2】（2 分）

使用说明中的术语，给出图中 D1～D2 所对应的数据存储名称。

【问题 3】（6 分）

使用说明和图中的术语，给出图 1-3 中加工 P1～P3 的名称。

【问题 4】（4 分）

解释说明图 1-2 和图 1-3 是否保持平衡，若不平衡请按如下格式补充图 1-3 中数据流的名称以及数据流的起点或终点，使其平衡（使用说明中的术语或图中符号）。

数据流名称	起点	终点

试题一分析

本题考查数据流图（DFD）的应用，采用结构化方法进行系统分析与设计，是一道传统题目，要求考生细心分析题目中所描述的内容。

DFD 是一种便于用户理解、分析系统数据流程的图形化建模工具，是系统逻辑模型的重要组成部分。

【问题 1】

本问题考查顶层 DFD。顶层 DFD 一般用来确定系统边界，将待开发系统看作一个加工，因此图中只有唯一的一个处理和一些外部实体，以及这两者之间的输入输出数据流。外部实体可以是使用系统的用户，也可以是为系统提供输入或接收系统输出的外部

系统。本问题要求根据描述确定图中的外部实体。应仔细分析题目中描述，并结合已经在顶层数据流图中给出的数据流进行分析。从题目的说明中可以看出，与系统的交互者包括应聘者、部门经理和工资系统。分析说明中的描述可知，应聘者提供自身信息，并接收系统验证合格后的致谢信息等。部门经理设置职位要求。对录用者而言，将其录用职位和信息发送给工资系统。对应图 1-1 中数据流和实体的对应关系，可知 E1 为应聘者，E2 为部门经理，E3 为工资系统。

【问题 2】

本问题考查 DFD 中数据存储的确定。本题中涉及的数据存储只有 2 个，一个是存储未被录用的应聘者信息，即未录用的应聘者表；另一个是存储对录用的应聘者进行职位安排评价的评价结果，即评价结果表。可以确定图 1-2 中 D1 和 D2 为未录用的应聘者表和评价结果表，因为有一个处理与这两个数据存储相关，需要再对应图 1-3，可确认 D1 为未录用的应聘者表，D2 为评价结果表。

【问题 3】

本问题考查 1 层 DFD 中缺失的处理。从说明（1）中接受申请的描述功能，需先对应聘者信息进行验证，受理验证合格的申请，可知缺失的处理 P1 为验证信息。说明（2）中，根据职位要求，审查已经受理的申请，对录用者进行职位安排评价，可知缺失的处理 P2 为审查申请，P3 为职位安排评价。

【问题 4】

本问题考查绘制分层 DFD 时的注意事项。在分层 DFD 中，需要保持父图与子图的平衡。即父图中某加工的输入输出数据流必须与其子图的输入输出数据流在数量和名字上相同，或者父图的一个输入（或输出）数据流对应于子图中几个输入（或输出）数据流，而子图中组成这些数据流的数据项全体正好是父图中的这一个数据流。

本题中，图 1-2 中加工的输入输出流与其子图 1-3 中的输入输出流的数量不同。也无须将父图中一条数据流分解成子图中多条数据流，因此，补充子图中缺失的输入或输出数据流：录用职位、已受理的申请、谢绝决策。

参考答案

【问题 1】

E1：应聘者　　E2：部门经理　　E3：工资系统

【问题 2】

D1：未录用的应聘者表　　D2：评价结果表

【问题 3】

P1：验证信息　　P2：审查申请　　P3：职位安排评价

【问题 4】

不平衡。图 1-2 中加工的输入输出流与其子图 1-3 中的输入输出流的数量不同。

数据流名称	起　点
录用职位	P3 或 2.3 职位安排评价
已受理的申请	1.2 受理申请
谢绝决策	2.2 谢绝应聘者

试题二（共 15 分）

阅读下列说明，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某物流公司为了整合上游供应商与下游客户，缩短物流过程，降低产品库存，需要构建一个信息系统以方便管理其业务运作活动。

【需求分析结果】

（1）物流公司包含若干部门，部门信息包括部门号、部门名称、经理、电话和邮箱。一个部门可以有多名员工处理部门的日常事务，每名员工只能在一个部门工作。每个部门有一名经理，只需负责管理本部门的事务和人员。

（2）员工信息包括员工号、姓名、职位、电话号码和工资；其中，职位包括：经理、业务员等。业务员根据托运申请负责安排承运货物事宜，例如：装货时间、到达时间等。一个业务员可以安排多个托运申请，但一个托运申请只由一个业务员处理。

（3）客户信息包括客户号、单位名称、通信地址、所属省份、联系人、联系电话、银行账号，其中，客户号唯一标识客户信息的每一个元组。每当客户要进行货物托运时，先要提出货物托运申请。托运申请信息包括申请号、客户号、货物名称、数量、运费、出发地、目的地。其中，一个申请号对应唯一的一个托运申请；一个客户可以有多个货物托运申请，但一个托运申请对应唯一的一个客户号。

【概念模型设计】

根据需求阶段收集的信息，设计的实体联系图和关系模式（不完整）如图 2-1 所示。

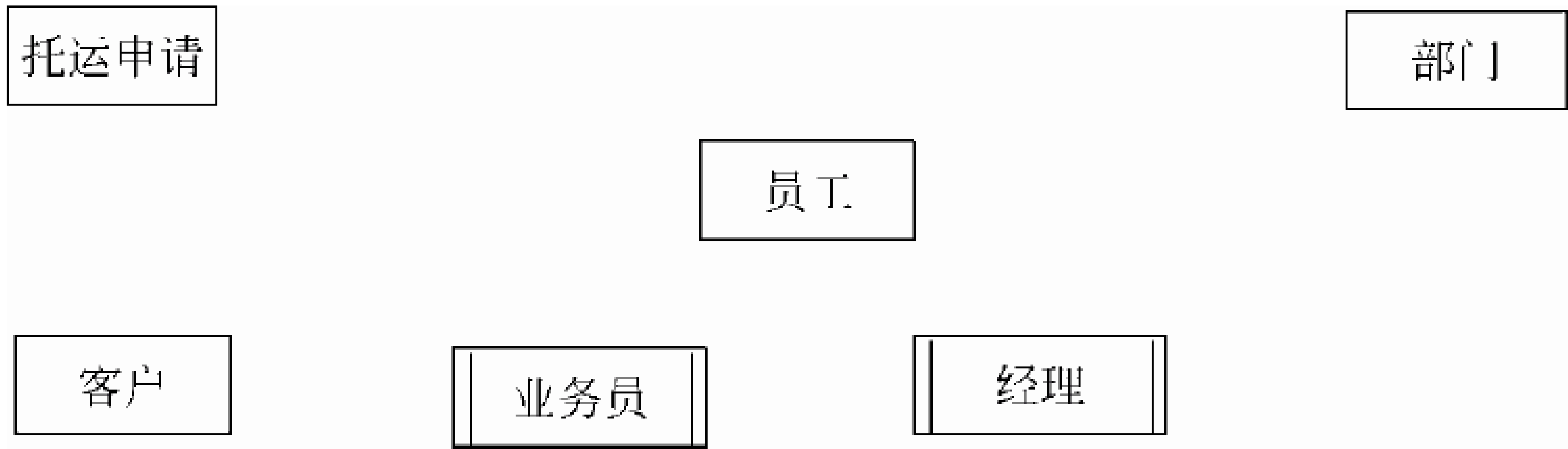


图 2-1　实体联系图

【关系模式设计】

部门（部门号，部门名称，经理，电话，邮箱）

员工（员工号，姓名，职位，电话号码，工资，__（a）__）

客户（__（b）__，单位名称，通信地址，所属省份，联系人，联系电话，银行账号）

托运申请（＿＿＿（c）＿＿＿，货物名称，数量，运费，出发地，目的地）

安排承运（＿＿＿（d）＿＿＿，装货时间，到达时间，业务员）

【问题 1】（5 分）

根据问题描述，补充四个联系、联系的类型，以及实体与子实体的联系，完善图 2-1 所示的实体联系图。

【问题 2】（8 分）

根据实体联系图，将关系模式中的空（a）～（d）补充完整。分别指出部门、员工和安排承运关系模式的主键和外键。

【问题 3】（2 分）

若系统新增需求描述如下：

为了数据库信息的安全性，公司要求对数据库操作设置权限管理功能，当员工登录系统时，系统需要检查员工的权限。权限的设置人是部门经理。为满足上述需要，应如何修改（或补充）图 2-1 所示的实体联系图，请给出修改后的实体联系图和关系模式。

试题二分析

本题考查数据库系统中实体联系模型（E-R 模型）和关系模式设计方面的应用知识。

【问题 1】

两个实体集之间的联系类型分为三类：一对一（1∶1）联系、一对多（1∶n）联系和多对多（m∶n）联系。

根据题意，每名员工只能在一个部门工作，所以部门和员工之间有一个 1∶n 的“所属”联系；由于每个部门有一名经理，只需负责管理本部门的事务和人员，因此部门和经理之间有一个 1∶1 的“管理”联系；由于一个业务员可以安排多个托运申请，但一个托运申请只由一个业务员处理，故业务员和托运申请之间有一个 1∶n 的“托运”联系；又由于一个客户可以有多个货物托运申请，但一个托运申请对应唯一的一个客户号，故客户和托运申请之间有一个 1∶n 的“申请”联系。

根据上述分析，完善图 2-1 所示的实体联系图可参见参考答案。

【问题 2】

根据题意，部门和员工之间有一个 1∶n 的“所属”联系需要将一端的码并入多端，故员工关系模式中的空（a）应填写部门号；在客户关系模式中，客户号为主键，故空（b）应填写客户号；在托运申请关系模式中，申请号、客户号为主键，故空（c）应填写申请号、客户号；又由于一个业务员可以安排多个托运申请，但一个托运申请只由一个业务员处理，因此在安排承运关系模式中，申请号为主键，故空（d）应填写申请号。

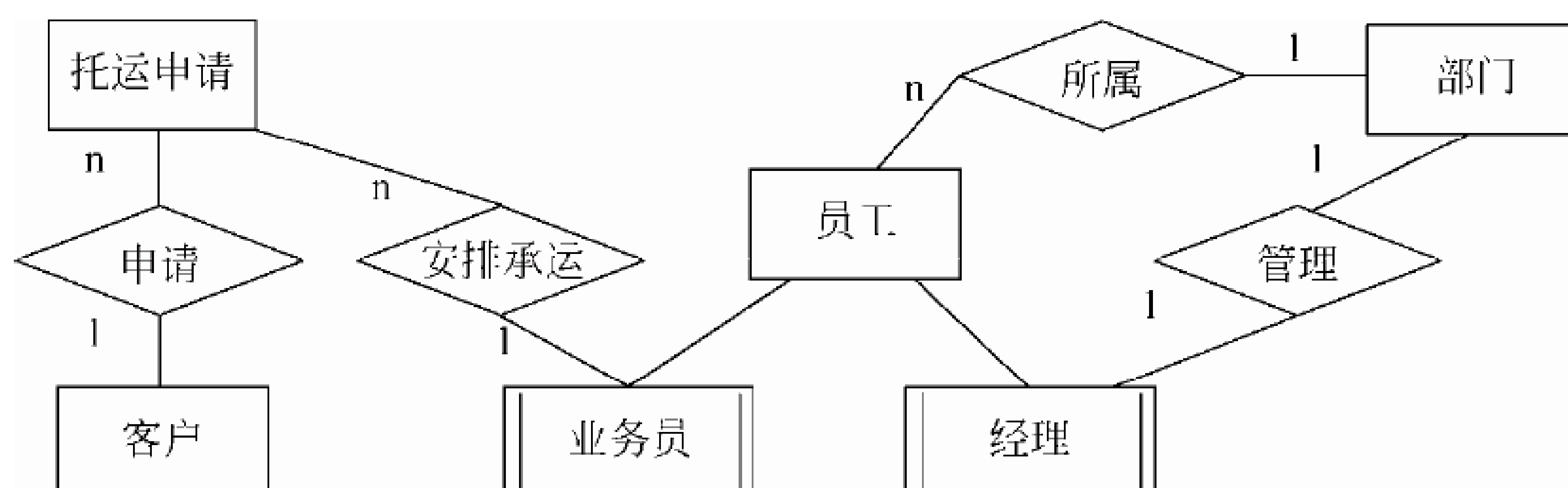
部门关系模式中的部门号为主键，经理为外键；因为经理来自员工关系。员工关系模式中的员工号为主键，部门号为外键，因为部门号来自部门关系。安排承运关系模式中的申请号为主键，业务员为外键，因为业务员来自员工关系。

【问题 3】

根据题意，权限的设置人是部门经理，因此，需要建立一个权限关系模式，以及经理到权限之间的 1：n 的“设置”联系。修改后的实体联系图和关系模式参见参考答案。

参考答案

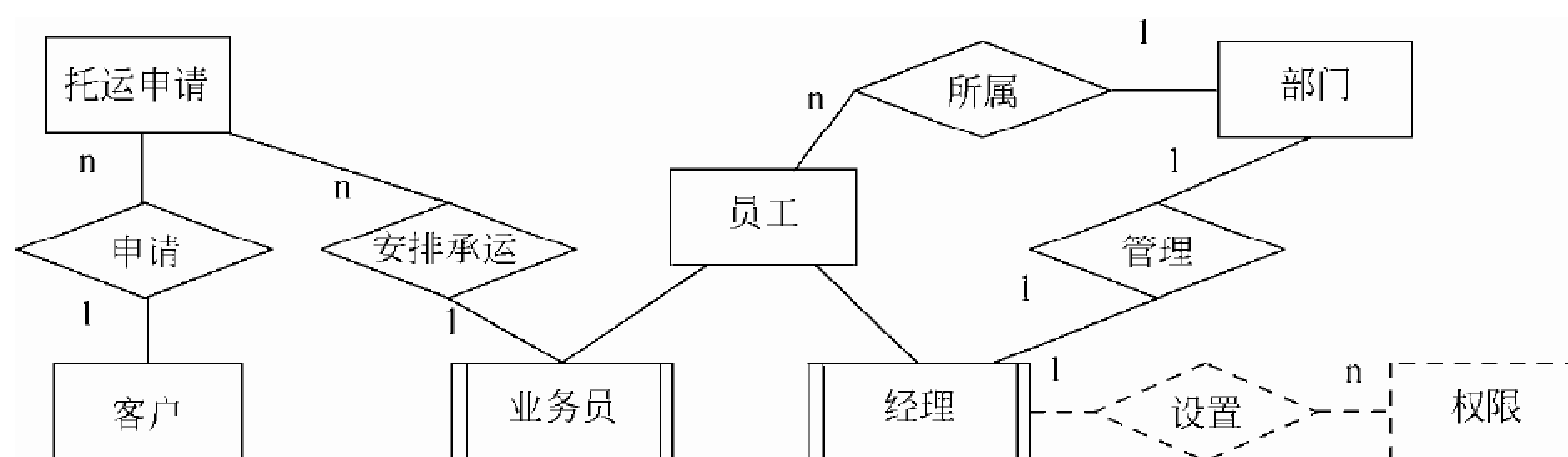
【问题 1】



【问题 2】

（a）	部门号	
（b）	客户号	
（c）	申请号，客户号	
（d）	申请号	
部门	主键：部门号	外键：经理
员工	主键：员工号	外键：部门号
安排承运	主键：申请号	外键：业务员

【问题 3】



关系模式：

权限（<u>员工号</u>，权限，设置人）

或　权限（<u>员工号</u>，权限，部门经理）

试题三（共 15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

Pay&Drive 系统（开多少付多少）能够根据驾驶里程自动计算应付的费用。

系统中存储了特定区域的道路交通网的信息。道路交通网由若干个路段（Road Segment）构成，每个路段由两个地理坐标点（Node）标定，其里程数（Distance）是已知的。在某些地理坐标点上安装了访问控制（Access Control）设备，可以自动扫描行驶卡（Card）。行程（Trajectory）由一组连续的路段构成。行程的起点（Entry）和终点（Exit）都装有访问控制设备。

系统提供了 3 种行驶卡。常规卡（Regular Card）有效期（Valid Period）为一年，可以在整个道路交通网内使用。季卡（Season Card）有效期为三个月，可以在整个道路交通网内使用。单次卡（Minitrip Card）在指定的行程内使用，且只能使用一次。其中，季卡和单次卡都是预付卡（Prepaid Card），需要客户（Customer）预存一定的费用。

系统的主要功能有：客户注册、申请行驶卡、使用行驶卡行驶等。

使用常规卡行驶，在进入行程起点时，系统记录行程起点、进入时间（Date Of Entry）等信息。在到达行程终点时，系统根据行驶的里程数和所持卡的里程单价（Unit Price）计算应付费用，并打印费用单（Invoice）。

季卡的使用流程与常规卡类似，但是不需要打印费用单，系统自动从卡中扣除应付费用。

单次卡的使用流程与季卡类似，但还需要在行程的起点和终点上检查行驶路线是否符合该卡所规定的行驶路线。

现采用面向对象方法开发该系统，使用 UML 进行建模。构建出的用例图和类图分别如图 3-1 和图 3-2 所示。

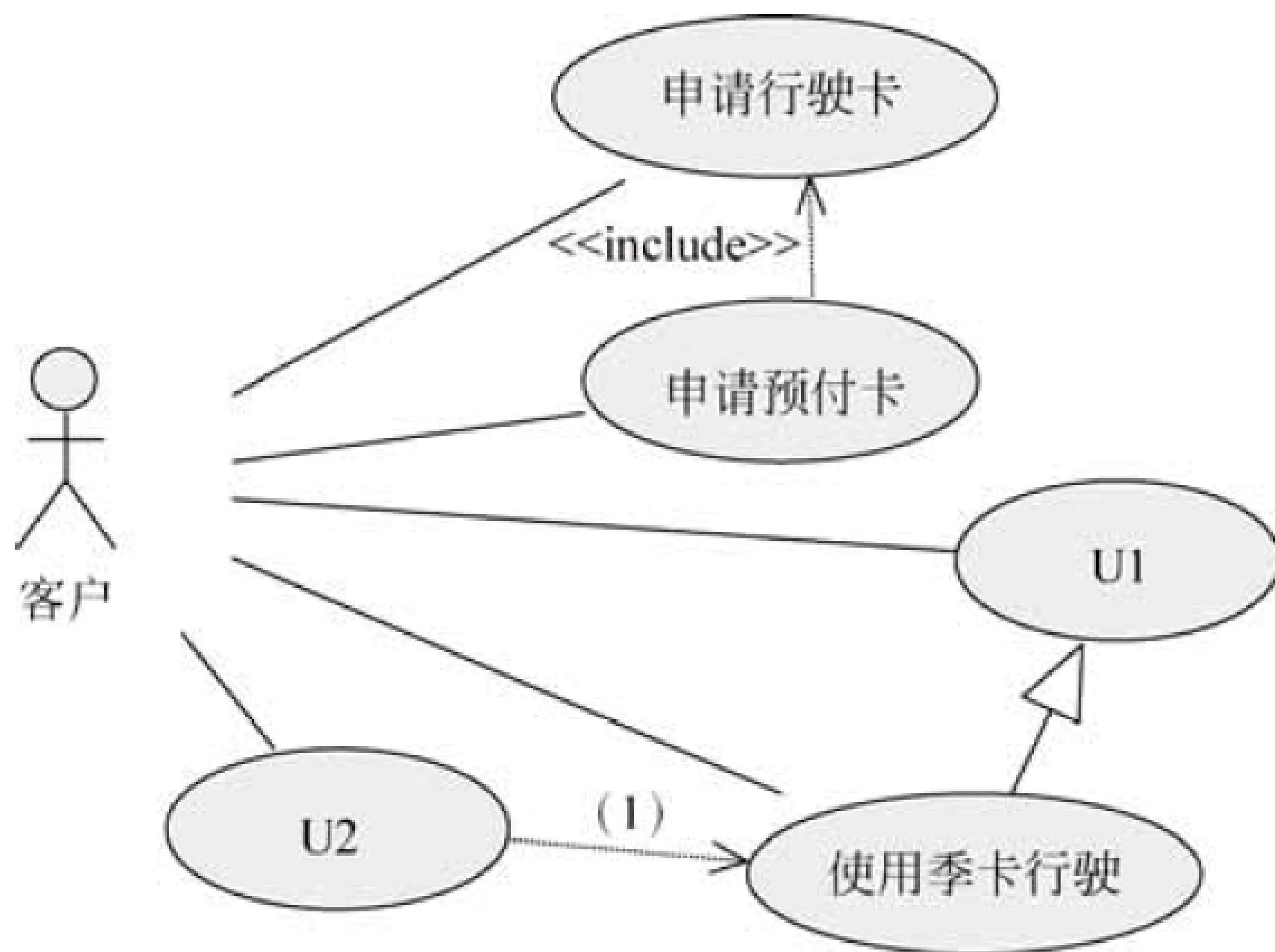


图 3-1　用例图

【问题 1】（4 分）

根据说明中的描述，给出图 3-1 中 U1 和 U2 所对应的用例，以及（1）所对应的

关系。

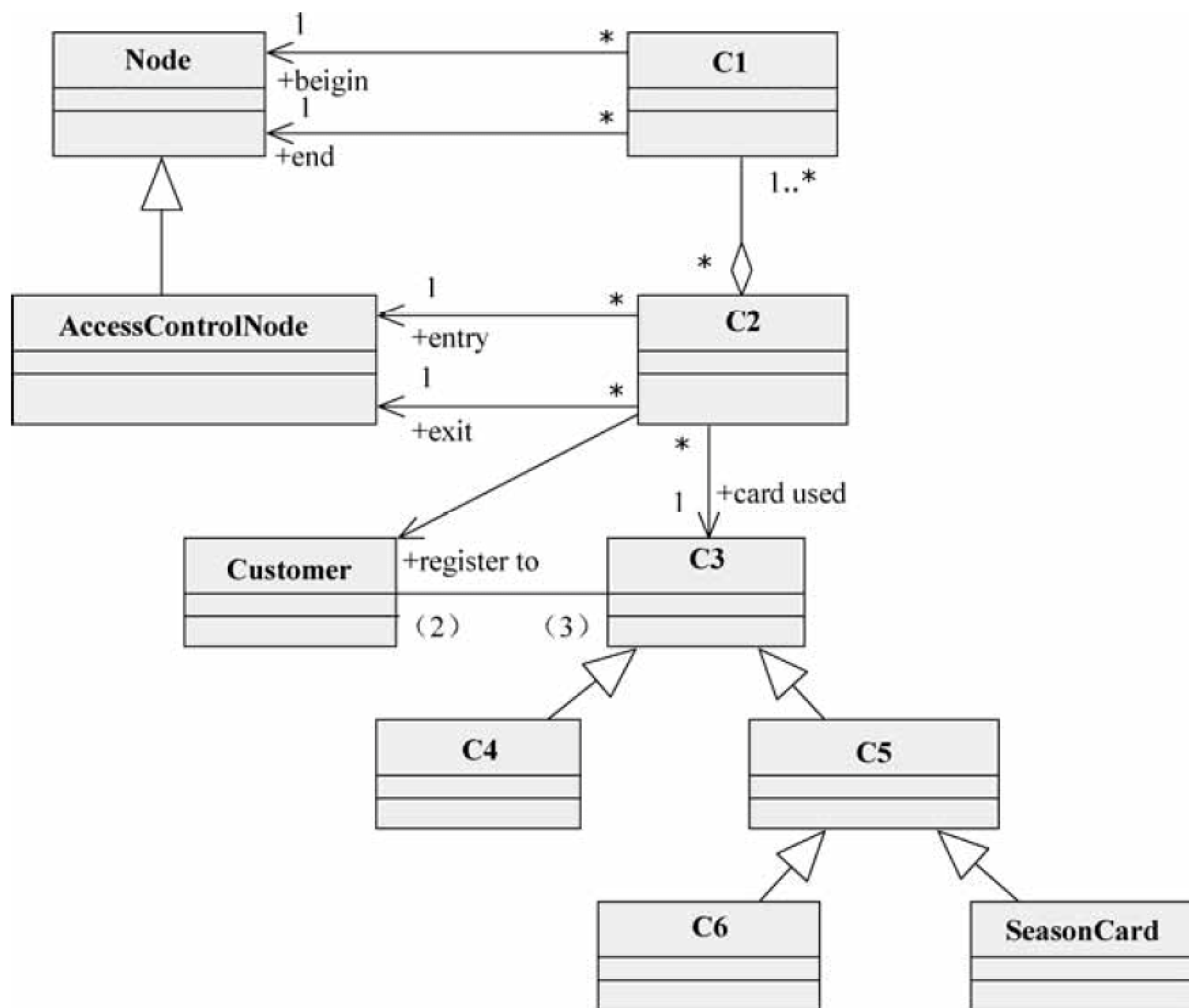


图 3-2 类图

【问题 2】（8 分）

根据说明中的描述，给出图 3-2 中缺少的 C1～C6 所对应的类名以及（2）～（3）处所对应的多重度（类名使用说明中给出的英文词汇）。

【问题 3】（3 分）

根据说明中的描述，给出 Road Segment、Trajectory 和 Card 所对应的类的关键属性（属性名使用说明中给出的英文词汇）。

试题三分析

本题属于经典的考题，主要考查面向对象分析方法以及 UML 的用例图和类图的相关知识。

【问题 1】

本问题要求将图 3-1 所给出的用例图补充完整。用例图的构成要素有：参与者、用例以及用例之间的关系。图中缺少了两个用例，以及一个用例关系。解答此题时，首先应从说明中找到所有的用例。

用例表示系统的一个单一业务功能。从题目的描述中可以看出，系统的主要功能就是申请行驶卡，以及使用行驶卡行驶。由于行驶卡分为三种，所以在说明中详细描述了

三种行驶卡的使用方法。再结合用例图来看，缺少的两个用例与用例“使用季卡行驶”有关联关系，由此可以推断出，需要补充的这两个用例必定与另外两种行驶卡相关，分别为“使用常规卡行驶”和“使用单次卡行驶”。

下面需要解决的问题是这两个用例与 U1 和 U2 的对应关系。这就需要仔细考查一下用例图所给出的用例关系。由图 3-1 可知，U1 和“使用季卡行驶”之间是泛化（generalization）关系。当多个用例共同拥有一种类似的结构和行为时，可以将它们的共性抽象为父用例，其他的用例作为泛化关系中的子用例。在用例的泛化关系中，子用例是父用例的一种特殊形式，子用例继承了父用例所有的结构、行为和关系。根据说明中的“季卡的使用流程与常规卡类似，但是不需要打印费用单，系统自动从卡中扣除应付费用”可知，U1 应该对应着用例“使用常规卡行驶”。由此不难得出 U2 对应着用例“用单次卡行驶”。

现在图中只剩下（1）处的用例关系没有确定。用例之间的关系在用例图上只有三种：包含（include）、扩展（extend）和泛化（generalization）。

包含关系是指当多个用例中存在相同事件流时，可以把这些公共事件流抽象成为公共用例，这个公共用例称为抽象用例，而原始用例称为基础用例。基础用例和抽象用例之间是包含关系。

如果一个用例明显地混合了两种或两种以上的不同场景，则可以将这个用例分为一个基本用例和多个扩展用例。扩展关系用“<<extend>>”表示，箭头指向基本用例。

包含关系和扩展关系的区别在于，抽象用例中的事件流一定要插入到基本用例中去，并且插入点只有一个，通常抽象用例不能脱离基本用例而独立存在。扩展用例的事件流往往可以抽象为基本用例的备选事件流，在扩展关系中，可以根据一定的条件来决定是否将扩展用例的事件流插入到基本用例的事件流中，并且插入点可以有多个。

根据以上分析可知，（1）处的用例关系选择“<<extend>>”最为合适。

【问题 2】

本问题考查的是类图建模。解题的重点在于根据类图中提供的类及类之间的关联关系，推断出剩余的类。

可以先观察一下类图。可以看到，需要补充的类基本上集中在两个结构上：聚集结构（类 C1 和 C2）以及继承结构（类 C3～C6）。继承结构是比较容易辨识的类之间的关联关系，图上给出了其中的一个子类 SeasonCard。以这个类为线索，回到说明中寻找与类 SeasonCard 相关的其他类。从说明中可知，“系统提供了 3 种卡”，常规卡、季卡、单次卡，而“季卡和单次卡都是预付卡”。这些描述暗示，“季卡”“单次卡”与“预付卡”之间存在着特殊/一般关系，即“is-a”关系，这是继承结构的典型标志。由此可以得出类 C5 和 C6 应该分别对应 PrepaidCard（预付卡）和 MinitripCard（单次卡）。根据 C5 和 C6 所对应的类，可以推断出，C4 和 C3 必定也是与行驶卡相关的类。三种卡中，已经有两种卡有了对应的类，还剩下一种卡即“常规卡”。而“常规卡”只能是与“预付卡”

同层次的概念，所以只能对应于C4，C3表示的是能代表所有这几种卡的公共概念。所以C3和C4应分别对应于Card和RegularCard。确定了C3之后，就可以识别出（2）和（3）处的多重度。Customer和Card之间是持有和被持有的关系，由于系统中只有3种卡，所以一个客户最多只能有3种卡，所以（3）处应填1..3。而对于任何一张卡来说，只能有唯一地一个所属人，因此（2）处应填1。

现在还剩下类C1和C2没有确定。由于这两个类之间是聚集关系，所以需要在说明中寻找具有“部分-整体”关系的概念。由说明中的“行程（Trajectory）由一组连续的路段构成”可知，C1和C2应分别对应于RoadSegment和Trajectory。

【问题3】

本问题考查类的关键属性的识别。由说明中给出的描述可知，类RoadSegment的属性至少应包括Distance；类Trajectory的属性至少应包括Entry、Exit和DateOfEntry；类Card的属性至少应包括UnitPrice、ValidPeriod。

参考答案

【问题1】

U1：使用常规卡行驶　　U2：使用单次卡行驶　　（1）：extend

【问题2】

C1：RoadSegment　　C2：Trajectory　　C3：Card

C4：RegularCard　　C5：PrepaidCard　　C6：MinitripCard

（2）1　　（3）1..3

【问题3】

RoadSegment的属性：Distance

Trajectory的属性：Entry、Exit、DateOfEntry

Card的属性：UnitPrice、ValidPeriod

试题四（共15分）

阅读下列说明和C代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

设某一机器由n个部件组成，每一个部件都可以从m个不同的供应商处购得。供应商j供应的部件i具有重量w_{ij}和价格c_{ij}。设计一个算法，求解总价格不超过上限cc的最小重量的机器组成。

采用回溯法来求解该问题：

首先定义解空间。解空间由长度为n的向量组成，其中每个分量取值来自集合{1,2,…,m}，将解空间用树形结构表示。

接着从根结点开始，以深度优先的方式搜索整个解空间。从根结点开始，根结点成为活结点，同时也成为当前的扩展结点。向纵深方向考虑第一个部件从第一个供应商处购买，得到一个新结点。判断当前的机器价格（c_{11}）是否超过上限（cc），重量（w_{11}）

是否比当前已知的解（最小重量）大，若是，应回溯至最近的一个活结点；若否，则该新结点成为活结点，同时也成为当前的扩展结点，根结点不再是扩展结点。继续向纵深方向考虑第二个部件从第一个供应商处购买，得到一个新结点。同样判断当前的机器价格（$c_{11}+c_{21}$）是否超过上限（cc），重量（$w_{11}+w_{21}$）是否比当前已知的解（最小重量）大。若是，应回溯至最近的一个活结点；若否，则该新结点成为活结点，同时也成为当前的扩展结点，原来的结点不再是扩展结点。以这种方式递归地在解空间中搜索，直到找到所要求的解或者解空间中已无活结点为止。

【C 代码】

下面是该算法的 C 语言实现。

（1）变量说明

n：机器的部件数

m：供应商数

cc：价格上限

w[][]：二维数组，w[i][j]表示第 j 个供应商供应的第 i 个部件的重量

c[][]：二维数组，c[i][j]表示第 j 个供应商供应的第 i 个部件的价格

bestW：满足价格上限约束条件的最小机器重量

bestC：最小重量机器的价格

bestX[]：最优解，一维数组，bestX[i]表示第 i 个部件来自哪个供应商

cw：搜索过程中机器的重量

cp：搜索过程中机器的价格

x[]：搜索过程中产生的解，x[i]表示第 i 个部件来自哪个供应商

i：当前考虑的部件，从 0 到 n–1

j：循环变量

（2）函数 backtrack

```
int n = 3;
int m = 3;
int cc = 4;
int w[3][3] = {{1,2,3},{3,2,1},{2,2,2}};
int c[3][3] = {{1,2,3},{3,2,1},{2,2,2}};
int bestW = 8;
int bestC = 0;
int bestX[3] = {0,0,0};
int cw = 0;
int cp = 0;
int x[3] = {0,0,0};
int backtrack(int i){
```

```
        int j = 0;
        int found = 0;
        if(i > n - 1){  /*得到问题解*/
            bestW = cw;
            bestC = cp;
            for(j = 0; j < n; j++){
                ______(1)______;
            }
    return 1;
        }
        if(cp <= cc){ /*有解*/
            found = 1;
        }
        for(j = 0; ______(2)______; j++){
            /*第 i 个部件从第 j 个供应商购买*/
    ______(3)______;
            cw = cw + w[i][j];
            cp = cp + c[i][j];
    if(cp <= cc &&______(4)______){  /*深度搜索，扩展当前结点*/
                if(backtrack(i + 1)){   found = 1;  }
            }
            /*回溯*/
            cw = cw - w[i][j];
            ______(5)______ ;
        }
        return found;
    }
```

试题四分析

本题考查算法的设计和分析技术中的回溯法。

回溯法是一种系统搜索问题解的方法，在包含问题所有解的解空间树中，按照深度优先的策略，从根结点出发搜索解空间树。算法在到达解空间树的任一结点时，总是先判断该结点是否肯定不包含问题的解。若肯定不包含，则跳过对以该结点为根的子树的系统搜索，逐层向其祖先结点回溯；否则进入该子树，继续按深度优先的策略进行搜索。回溯法在求问题的最优解时，要回溯到根，且根结点的所有子树都已经被搜索遍才结束。

根据上述思想和题干说明，对实例：部件数 n＝3，厂商数 m=3，具体的重量和价格如表 4-1 所示。

表 4-1　每个部件的重量和价格

		n					
		1		2		3	
		w_1	c_1	w_2	c_2	w_3	c_3
m	1	1	1	2	2	3	3
	2	3	3	2	2	1	1
	3	2	2	2	2	2	2

构造该实例的解空间树如图 4-1 所示。

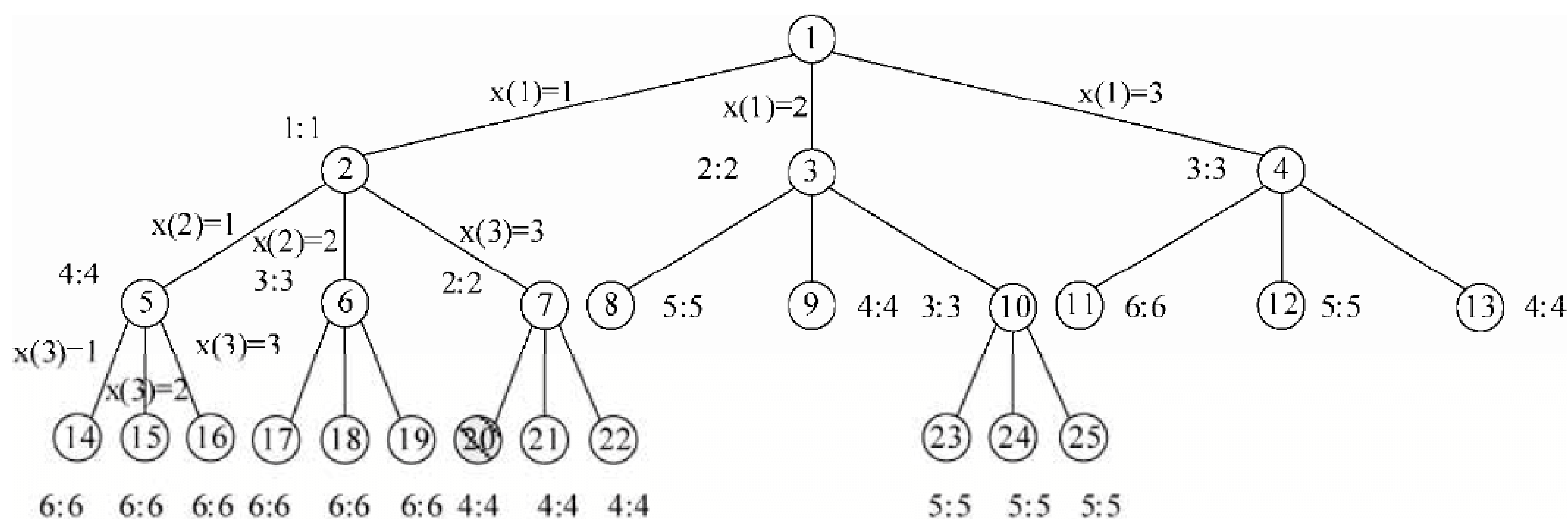


图 4-1　解空间树

图 4-1 中结点编号表示生成该结点的顺序。边上的编号表示哪个部件选择哪个厂商，如 x(2)=1，表示第 2 个部件来自厂商 1。结点旁边的两个数字表示当前解或部分解对应的重量和价格，如 2∶2 表示重量为 2，价格为 2。从图 4-1 可以看出，最优解是结点 20 表示的解，即 x(1)=1，x(2)=3，x(3)=1，即第 1 个部件来自厂商 1，第 2 个部件来自厂商 3，第 3 个部件来自厂商 1，总的价格和重量分别为 4 和 4。当然，本实例的最优解还可以是 x(1)=1，x(2)=3，x(3)=2 和 x(1)=1，x(2)=3，x(3)=3，分别对应解空间树上的 21 号和 22 号结点。

代码中的空（1）处是得到问题解之后，将搜索过程中产生的重量 cw、价格 cp 和解 x 放到最终重量 bestW、价格 bestC 和解 bestX 中，因此空格（1）处填写 bestX[j] = x[j]。空（2）处的 for 循环是考虑第 i 个部件选择哪个厂商，因此 j 从 0 到 m–1 依次检查，此处应填 j<m。对搜索过程中产生的重量 cw、价格 cp 和解 x 的值进行设置，因此空（3）处应填 x[i] = j ，表示第 i 个部件选择厂商 j。空（4）是判断当前结点是否要扩展，若当前获得的价格比目前最优解更优，且重量没有超过当前得到的最优重量，即 cp <= cc 且 cw < bestW，则扩展当前结点，否则回溯。在回溯过程中，需要把原来选择的部件的价格和重量从搜索过程中产生的重量 cw 和价格 cp 中去掉，因此空(5)应填 cp = cp - c[i][j]。

参考答案

（1）bestX[j] = x[j]　　（2）j < m　　（3）x[i] = j

（4）cw < bestW　　（5）cp = cp–c[i][j]

试题五（共 15 分）

阅读下列说明和 C++代码，将应填入__(n)__处的字句写在答题纸的对应栏内。

【说明】

某大型商场内安装了多个简易的纸巾售卖机，自动出售 2 元钱一包的纸巾，且每次仅售出一包纸巾。纸巾售卖机的状态图如图 5-1 所示。

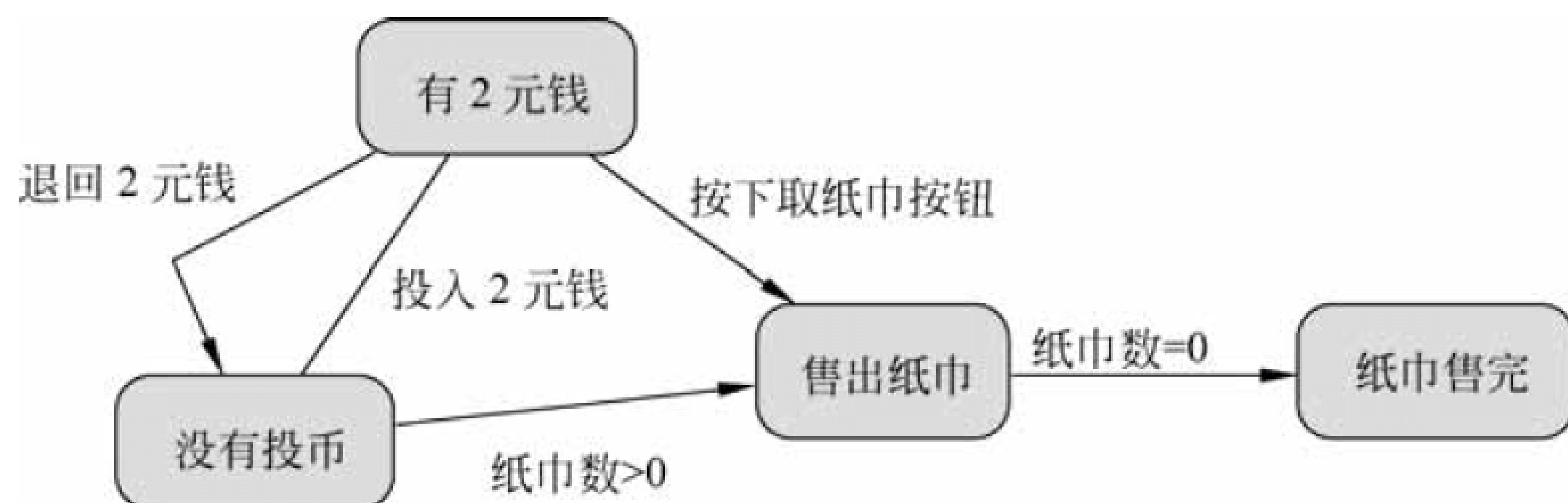


图 5-1 纸巾售卖机状态图

采用状态（State）模式来实现该纸巾售卖机，得到如图 5-2 所示的类图。其中类 State 为抽象类，定义了投币、退币、出纸巾等方法接口。类 SoldState、SoldOutState、NoQuarterState 和 HasQuarterState 分别对应图 5-1 中纸巾售卖机的 4 种状态：售出纸巾、纸巾售完、没有投币、有 2 元钱。

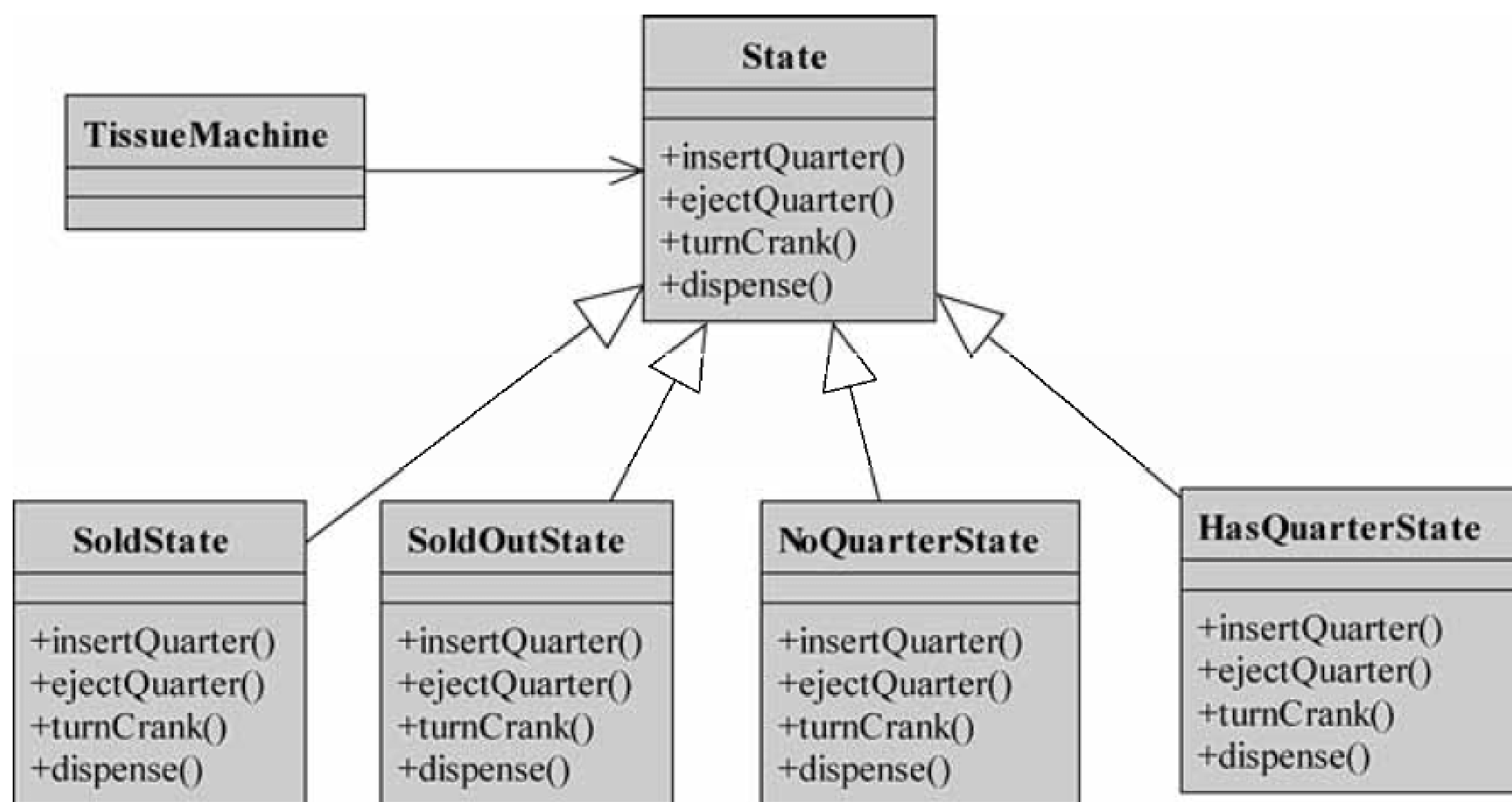


图 5-2 类图

【C++代码】

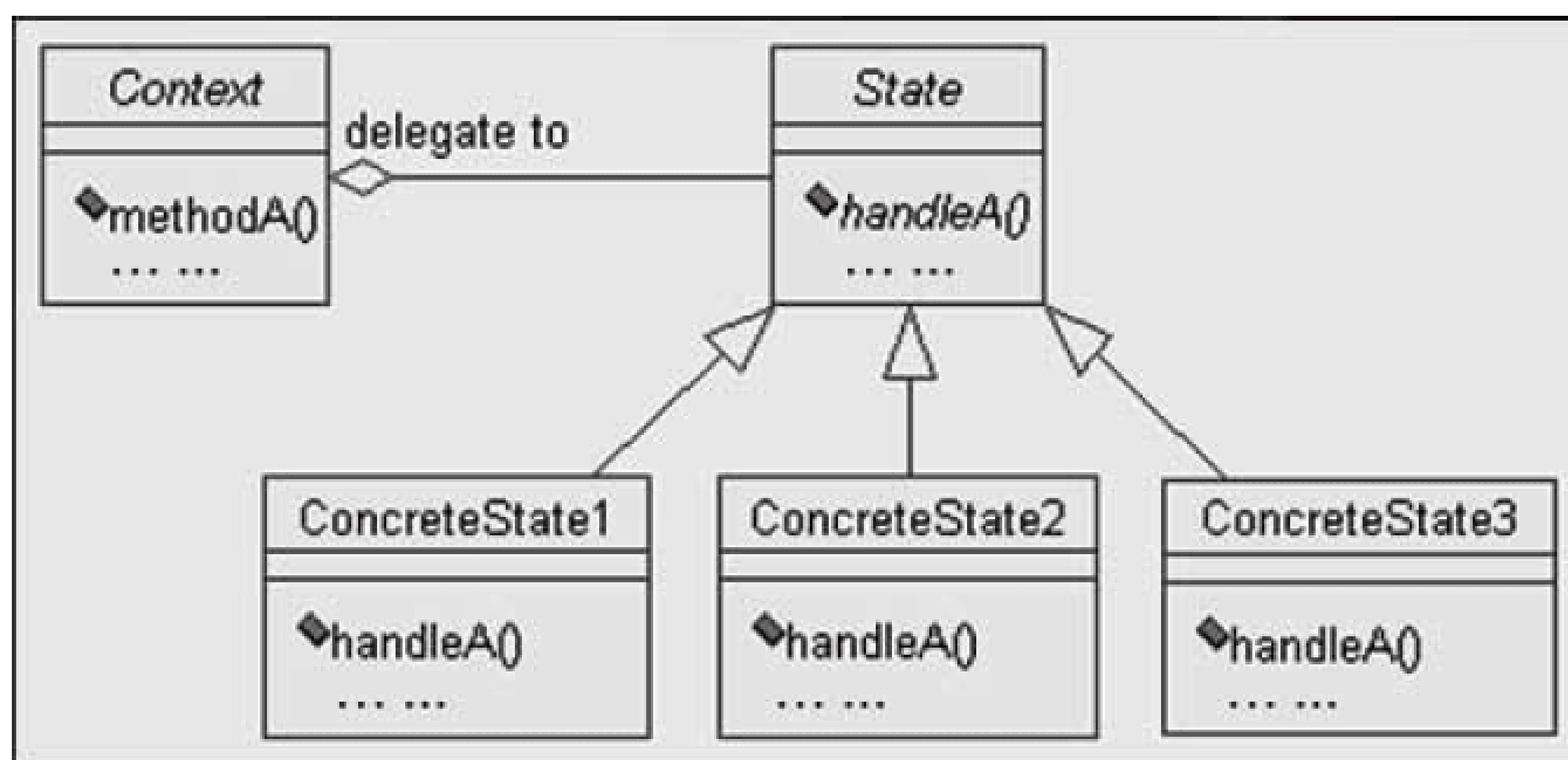
```
#include <iostream>
using namespace std;
// 以下为类的定义部分
class TissueMachine;                    // 类的提前引用
```

```
class State {
public:
   virtual void insertQuarter() = 0;//投币
   virtual void ejectQuarter() = 0; //退币
   virtual void turnCrank()= 0;     //按下“出纸巾”按钮
   virtual void dispense() = 0;     //出纸巾
};
/* 类 SoldOutState、NoQuarterState、HasQuarterState、SoldState 的定义省略，
每个类中均定义了私有数据成员 TissueMachine* tissueMachine; */
class TissueMachine {
private:
   ___(1)___ *soldOutState, *noQuarterState, *hasQuarterState,*soldState,
*state ;
   int count;                       //纸巾数
public:
  TissueMachine(int numbers);
  void setState(State* state);
  State* getHasQuarterState();
  State* getNoQuarterState();
  State* getSoldState();
  State* getSoldOutState();
  int getCount();
  //其余代码省略
};
//以下为类的实现部分
void NoQuarterState ::insertQuarter() {
tissueMachine->setState(___(2)___);
}
void HasQuarterState ::ejectQuarter() {
tissueMachine->setState(___(3)___);
}
void SoldState ::dispense() {
    if(tissueMachine->getCount() > 0) {
      tissueMachine->setState(___(4)___);
}
else {
      tissueMachine->setState(___(5)___);
}
}  //其余代码省略
```

试题五分析

本题考查状态（State）模式的概念及应用。

状态模式是一种对象的行为型模式，允许一个对象在其内部状态改变时改变它的行为，对象看起来似乎修改了它的类。状态模式的类图如下所示：



状态模式主要解决的是控制一个对象转换的条件表达式过于复杂的情况。把状态的判断逻辑转移到表示不同状态的一系列类当中，可以把复杂的判断逻辑简化。状态模式的好处是将与特定状态相关的行为局部化，并且将不同状态的行为分割开来。

题目利用状态模式来实现一个简易的纸巾售卖机。售卖机的状态转换图已经在题目中给出，类 SoldState、SoldOutState、NoQuarterState 和 HasQuarterState 分别用来表示售卖机的 4 种不同状态，对应于状态模式中的 ConcreteState1，…ConcreteStateN。题目所设置的填空，主要集中在状态转换上。因此解答该题时，要求在理解状态模式内涵的基础上，依据纸巾售卖机的状态转换原则，给出正确的状态设置。

空（1）出现在类 TissueMachine 的数据成员定义部分。状态模式封装了状态的转换过程，但是它需要枚举可能的状态，因此需要确定状态种类。因此在类 TissueMachine 中需定义出所有可能的状态对象。根据所给出的对象名称及说明中的描述，可知（1）处应填入的类名为 State。

空（2）～（5）都是与状态转换相关的，要求填写类 TissueMachine 中的方法 setState 在不同调用处的实际参数。根据方法的名称及调用方式，可以推断出这个方法的功能就是设置自动售卖机的当前状态。要填出这些空，只要对照图 5-1 的状态转换图，根据状态转换的条件确定出当前状态及下一状态即可。

空（2）出现在方法 insertQuarter 内，即给纸巾售卖机投入 2 元钱。根据状态图，“投入 2 元钱”之后，售卖机应转换到“有 2 元钱”的状态。“有 2 元钱”对应的状态的类为“HasQuarterState”，所以空（2）处应填类 HasQuarterState 的对象。由于 hasQuarterState 是类 TissueMachine 的私有数据成员，不能直接访问，所以只能通过调用相关的 get 方法来获取该对象。由此得出（2）应填 tissueMachine->getHasQuarterState()。

同理，空（3）表示的状态是从“有 2 元钱”状态，经历“退回 2 元钱”事件之后的状态，及“没有投币”状态。所以空（3）处应填 tissueMachine->getNoQuarterState()。

空（4）和（5）处分别表示卖出一包纸巾之后，售卖机应该转换到的下一个状态。这个跟售卖机中的纸巾数有关，如果还有纸巾，则转换到“没有投币”状态，如果没有纸巾了，则转换到“纸巾售完”状态，因此，空（4）处应填 tissueMachine->

getNoQuarterState()，空（5）处应填 tissueMachine->getSoldOutState()。

参考答案

（1）State

（2）tissueMachine->getHasQuarterState()

（3）tissueMachine->getNoQuarterState()

（4）tissueMachine->getNoQuarterState()

（5）tissueMachine->getSoldOutState()

试题六（共 15 分）

阅读下列说明和 Java 代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

某大型商场内安装了多个简易的纸巾售卖机，自动出售 2 元钱一包的纸巾，且每次仅售出一包纸巾。纸巾售卖机的状态图如图 6-1 所示。

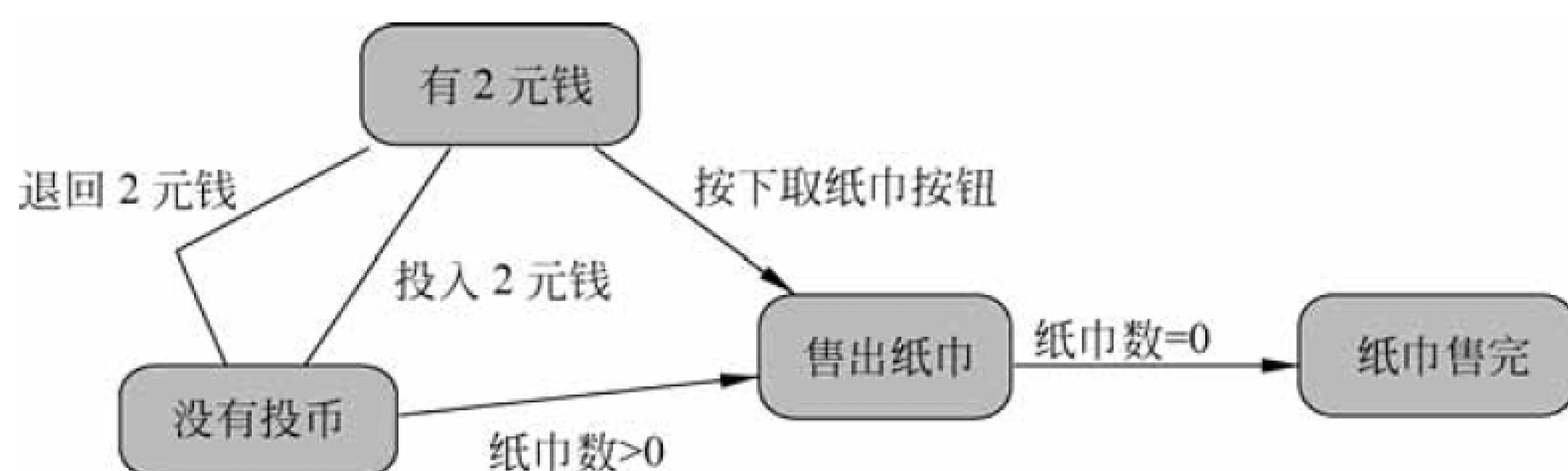


图 6-1　纸巾售卖机状态图

采用状态（State）模式来实现该纸巾售卖机，得到如图 6-2 所示的类图。其中类 State 为抽象类，定义了投币、退币、出纸巾等方法接口。类 SoldState、SoldOutState、NoQuarterState 和 HasQuarterState 分别对应图 6-1 中纸巾售卖机的 4 种状态：售出纸巾、纸巾售完、没有投币、有 2 元钱。

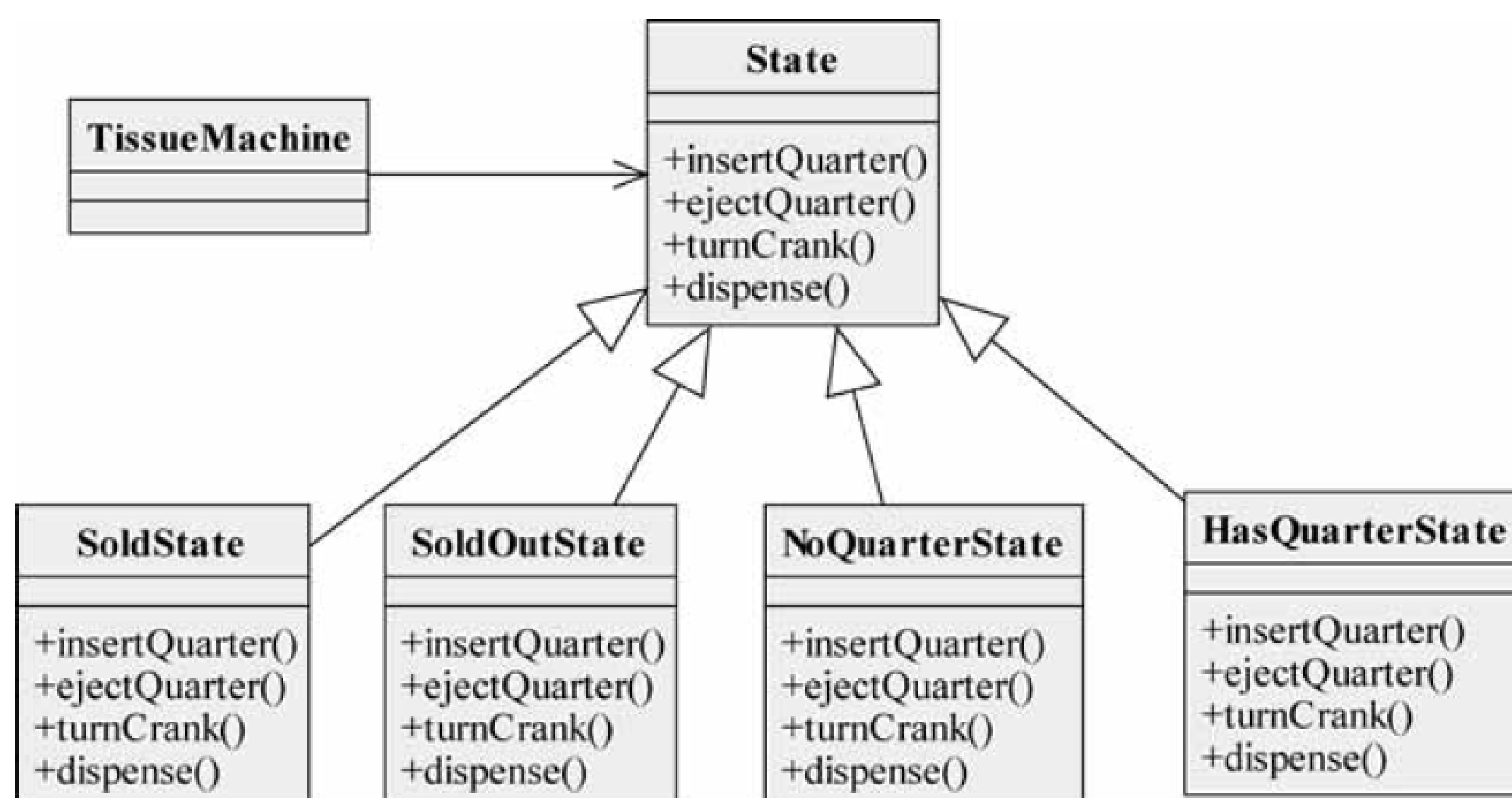


图 6-2　类图

【Java 代码】

```
import java.util.*;
interface State {
  public void insertQuarter();        //投币
  public void ejectQuarter();         //退币
  public void turnCrank();            //按下“出纸巾”按钮
  public void dispense();             //出纸巾
}
class TissueMachine {
  ___(1)___ soldOutState, noQuarterState, hasQuarterState, soldState,
state;
  state = soldOutState;
  int count = 0;                      //纸巾数
  public TissueMachine(int numbers) { /* 实现代码省略 */ }
public State getHasQuarterState() { return hasQuarterState; }
  public State getNoQuarterState() { return noQuarterState; }
  public State getSoldState()      { return soldState;       }
  public State getSoldOutState()   { return soldOutState;    }
  public int getCount()            { return count;           }
//其余代码省略
}

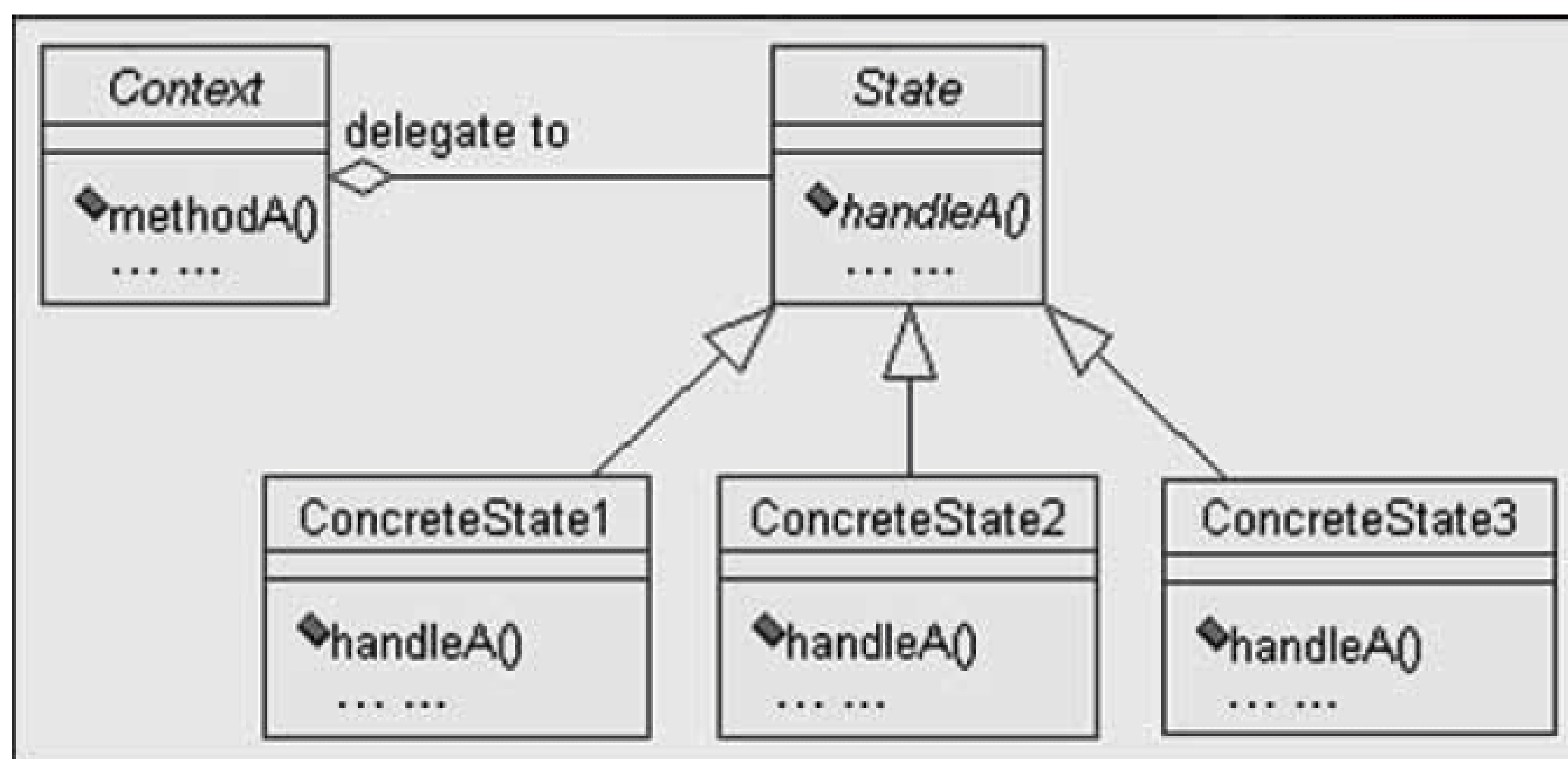
class NoQuarterState implements State {
  TissueMachine tissueMachine;
  public void insertQuarter() {
    tissueMachine.setState(___(2)___);
  }
//构造方法以及其余代码省略
}
class HasQuarterState implements State {
  TissueMachine tissueMachine;
  public void ejectQuarter() {
    tissueMachine.setState(___(3)___);
  }
//构造方法以及其余代码省略
}
class SoldState implements State {
  TissueMachine tissueMachine;
  public void dispense() {
    if(tissueMachine.getCount() > 0) {
```

```
        tissueMachine.setState(  (4)  );
    } else {
        tissueMachine.setState(  (5)  );  }
      }
    }
```

试题六分析

本题考查状态（State）模式的概念及应用。

状态模式是一种对象的行为型模式，允许一个对象在其内部状态改变时改变它的行为，对象看起来似乎修改了它的类。状态模式的类图如下所示：



状态模式主要解决的是控制一个对象转换的条件表达式过于复杂的情况。把状态的判断逻辑转移到表示不同状态的一系列类当中，可以把复杂的判断逻辑简化。状态模式的好处是将与特定状态相关的行为局部化，并且将不同状态的行为分割开来。

题目利用状态模式来实现一个简易的纸巾售卖机。售卖机的状态转换图已经在题目中给出，类 SoldState、SoldOutState、NoQuarterState 和 HasQuarterState 分别用来表示售卖机的 4 种不同状态，对应于状态模式中的 ConcreteState1，…ConcreteStateN。题目所设置的填空，主要集中在状态转换上。因此解答该题时，要求在理解状态模式内涵的基础上，依据纸巾售卖机的状态转换原则，给出正确的状态设置。

空（1）出现在类 TissueMachine 的数据成员定义部分。状态模式封装了状态的转换过程，但是它需要枚举可能的状态，因此需要实现确定状态种类。因此在类 TissueMachine 中需定义出所有可能的状态对象。根据所给出的对象名称及说明中的描述，可知（1）处应填入的类名为 State。

空（2）～（5）要求填写类 TissueMachine 中的方法 setState 在不同调用处的实际参数。这里的一个难点在于题目中没有显示地给出方法 setState 的原型及语义，这要求考生根据面向对象程序设计风格及说明中给出的应用场合来推断 setState 的内涵及原型，主要是确定其参数列表。

在面向对象程序设计中，为了做到封装，通常都会把数据成员定义为私有的。私有的数据成员对象不能直接访问，因此在类中都会提供 2 组访问私有数据成员的方法，分

别为 get…方法和 set…方法（…代表对应的数据成员名称）。get…方法表示获取私有数据成员的值，其返回值类型为对应的数据成员的类型；set…方法表示对数据成员进行赋值，所要赋的值通常通过参数传递进去，方法的返回值类型通常为 void。根据面向对象程序设计的这些特点，以及状态模式的内涵及应用场合，可以推断出 setState 方法的功能就是设置纸巾售卖机的当前状态。纸巾售卖机在任一时刻只能处于一个唯一的状态，由状态模式可知，纸巾售卖机的状态都是用状态对象表示的，由此就可以确定出，setState 方法的参数只要一个就可以了，就是表示纸巾售卖机下一状态的状态对象。

经过以上分析之后，可以明确空（2）～（5）空所填的内容都应与状态转换相关。因此要填充这些空，只要对照图 5-1 的状态转换图，根据状态转换的条件确定出当前状态及下一状态即可。

空（2）出现在方法 insertQuarter 内，即给纸巾售卖机投入 2 元钱。根据状态图，“投入 2 元钱”之后，售卖机应转换到“有 2 元钱”的状态。“有 2 元钱”对应的状态的类为“HasQuarterState”，所以空（2）处应填写类 HasQuarterState 的对象。由此得出（2）应填 tissueMachine.getHasQuarterState()。

同理，空（3）表示的状态是从“有 2 元钱”状态，经历“退回 2 元钱”事件之后的状态，及“没有投币”状态。所以空（3）处应填 tissueMachine.getNoQuarterState()。

空（4）和（5）处分别表示卖出一包纸巾之后，售卖机应该转换到的下一个状态。这个跟售卖机中的纸巾数有关，如果还有纸巾，则转换到“没有投币”状态，如果没有纸巾了，则转换到“纸巾售完”状态，因此，空（4）处应填 tissueMachine.getNoQuarterState()，空（5）处应填 tissueMachine.getSoldOutState()。

参考答案

（1）State

（2）tissueMachine.getHasQuarterState()

（3）tissueMachine.getNoQuarterState()

（4）tissueMachine.getNoQuarterState()

（5）tissueMachine.getSoldOutState()

第 13 章　2012 上半年软件设计师上午试题分析与解答

试题（1）

位于 CPU 与主存之间的高速缓冲存储器（Cache）用于存放部分主存数据的拷贝，主存地址与 Cache 地址之间的转换工作由<u>（1）</u>完成。

（1）A. 硬件　　B. 软件　　C. 用户　　D. 程序员

试题（1）分析

本题考查高速缓冲存储器（Cache）的工作特点。

提供“高速缓存”的目的是为了让数据存取的速度适应 CPU 的处理速度，其基于的原理是内存中“程序执行与数据访问的局域性行为”，即一定程序执行时间和空间内，被访问的代码集中于一部分。为了充分发挥高速缓存的作用，不仅依靠“暂存刚刚访问过的数据”，还要使用硬件实现的指令预测与数据预取技术，即尽可能把将要使用的数据预先从内存中取到高速缓存中。

一般而言，主存使用DRAM技术，而 Cache 使用昂贵但较快速的SRAM技术。

目前微计算机上使用的AMD或Intel微处理器都在芯片内部集成了大小不等的数据高速缓存和指令高速缓存，通称为 L1 高速缓存（L1 Cache，即第一级片上高速缓冲存储器）；而比 L1 容量更大的 L2 高速缓存曾经被放在 CPU 外部（主板或者 CPU 接口卡上），但是现在已经成为 CPU 内部的标准组件；更昂贵的顶级家用和工作站CPU甚至会配备比 L2 高速缓存还要大的 L3 高速缓存。

参考答案

（1）A

试题（2）

内存单元按字节编址，地址 0000A000H～0000BFFFH 共有<u>（2）</u>个存储单元。

（2）A. 8192K　　B. 1024K　　C. 13K　　D. 8K

试题（2）分析

本题考查存储器的地址计算知识。

每个地址编号为一个存储单元（容量为 1 个字节），地址区间 0000A000H～0000BFFFH 共有 1FFF+1 个地址编号（即 2^{13}），1K=1024，因此该地址区间的存储单元数也就是 8K。

参考答案

（2）D

试题（3）

相联存储器按__（3）__访问。

（3）A. 地址　　B. 先入后出的方式
C. 内容　　D. 先入先出的方式

试题（3）分析

本题考查相联存储器的概念。

相联存储器是一种按内容访问的存储器。其工作原理就是把数据或数据的某一部分作为关键字，将该关键字与存储器中的每一单元进行比较，找出存储器中所有与关键字相同的数据字。

相联存储器可用在高速缓冲存储器中，在虚拟存储器中用来作段表、页表或快表存储器，还用在数据库和知识库中。

参考答案

（3）C

试题（4）

若 CPU 要执行的指令为：MOV　R1，#45（即将数值 45 传送到寄存器 R1 中），则该指令中采用的寻址方式为__（4）__。

（4）A. 直接寻址和立即寻址　　B. 寄存器寻址和立即寻址
C. 相对寻址和直接寻址　　D. 寄存器间接寻址和直接寻址

试题（4）分析

本题考查指令系统基础知识。

指令中的寻址方式就是如何对指令中的地址字段进行解释，以获得操作数的方法或获得程序转移地址的方法。常用的寻址方式有：

- 立即寻址。操作数就包含在指令中。
- 直接寻址。操作数存放在内存单元中，指令中直接给出操作数所在存储单元的地址。
- 寄存器寻址。操作数存放在某一寄存器中，指令中给出存放操作数的寄存器名。
- 寄存器间接寻址。操作数存放在内存单元中，操作数所在存储单元的地址在某个寄存器中。
- 间接寻址。指令中给出操作数地址的地址。
- 相对寻址。指令地址码给出的是一个偏移量（可正可负），操作数地址等于本条指令的地址加上该偏移量。
- 变址寻址。操作数地址等于变址寄存器的内容加偏移量。

题目给出的指令中，R1 是寄存器，属于寄存器寻址方式，45 是立即数，属于立即寻址方式。

参考答案

（4）B

试题（5）、（6）

一条指令的执行过程可以分解为取指、分析和执行三步，在取指时间 $t_{取指}=3\Delta t$、分析时间 $t_{分析}=2\Delta t$、执行时间 $t_{执行}=4\Delta t$ 的情况下，若按串行方式执行，则 10 条指令全部执行完需要 __（5）__ Δt。若按照流水方式执行，则执行完 10 条指令需要 __（6）__ Δt。

（5）A. 40　　B. 70　　C. 90　　D. 100

（6）A. 20　　B. 30　　C. 40　　D. 45

试题（5）、（6）分析

本题考查指令执行的流水化概念。

根据题目中给出的数据，每一条指令的执行过程需要 $9\Delta t$。在串行执行方式下，执行完一条指令后才开始执行下一条指令，10 条指令共耗时 $90\Delta t$。若按照流水方式执行，则在第 i+2 条指令处于执行阶段时就可以分析第 i+1 条指令，同时取第 i 条指令。由于指令的执行阶段所需时间最长（为 $4\Delta t$），因此指令开始流水执行后，每 $4\Delta t$ 将完成一条指令，所需时间为 $3\Delta t+2\Delta t+4\Delta t+4\Delta t\times 9=45\Delta t$。

参考答案

（5）C　（6）D

试题（7）

甲和乙要进行通信，甲对发送的消息附加了数字签名，乙收到该消息后利用 __（7）__ 验证该消息的真实性。

（7）A. 甲的公钥　　B. 甲的私钥　　C. 乙的公钥　　D. 乙的私钥

试题（7）分析

本题考查数字签名的概念。

数字签名（Digital Signature）技术是不对称加密算法的典型应用：数据源发送方使用自己的私钥对数据校验和（或）其他与数据内容有关的变量进行加密处理，完成对数据的合法“签名”，数据接收方则利用对方的公钥来解读收到的“数字签名”，并将解读结果用于对数据完整性的检验，以确认签名的合法性。数字签名主要的功能是保证信息传输的完整性、发送者的身份认证、防止交易中的抵赖发生。

参考答案

（7）A

试题（8）

在 Windows 系统中，默认权限最低的用户组是 __（8）__。

（8）A. everyone　　B. administrators　　C. power users　　D. users

试题（8）分析

本题考查 Windows 用户权限方面的知识。

在以上 4 个选项中，用户组默认权限由高到低的顺序是 administrators→power users →users→everyone

参考答案

（8）A

试题（9）

IIS 6.0 支持的身份验证安全机制有 4 种验证方法，其中安全级别最高的验证方法是__（9）__。

（9）A. 匿名身份验证　　B. 集成 Windows 身份验证
C. 基本身份验证　　D. 摘要式身份验证

试题（9）分析

本题考查 Windows IIS 服务中身份认证的基础知识。

Windows IIS 服务支持的身份认证方式有.NET Passport 身份验证、集成 Windows 身份验证、摘要式身份验证和基本身份验证。

- 集成 Windows 身份验证：以 Kerberos 票证的形式通过网络向用户发送身份验证信息，并提供较高的安全级别。Windows 集成身份验证使用 Kerberos 版本 5 和 NTLM 身份验证。
- 摘要式身份验证：将用户凭据作为 MD5 哈希或消息摘要在网络中进行传输，这样就无法根据哈希对原始用户名和密码进行解码。
- .NET Passport 身份验证：对 IIS 的请求必须在查询字符串或 Cookie 中包含有效的.NET Passport 凭据，提供了单一登录安全性，为用户提供对 Internet 上各种服务的访问权限。
- 基本身份验证：用户凭据以明文形式在网络中发送。这种形式提供的安全级别很低，因为几乎所有协议分析程序都能读取密码。

参考答案

（9）B

试题（10）

软件著作权的客体不包括__（10）__。

（10）A. 源程序　　B. 目标程序　　C. 软件文档　　D. 软件开发思想

试题（10）分析

软件著作权的客体是指著作权法保护的计算机软件，包括计算机程序及其相关文档。

计算机程序通常包括源程序和目标程序。

源程序（又称为源代码、源码）是采用计算机程序设计语言（如 C、Java 语言）编写的程序，需要转换成机器能直接识别和执行的形式才能在计算机上运行并得出结果。它具有可操作性、间接应用性和技术性等特点。

目标程序以二进制编码形式表示，是计算机或具有信息处理能力的装置能够识别和

执行的指令序列，能够直接指挥和控制计算机的各部件（如存储器、处理器、I/O 设备等）执行各项操作，从而实现一定的功能。它具有不可读性、不可修改性和面向机器性等特点。

源程序与目标程序就其逻辑功能而言不仅内容相同，而且表现形式相似，二者可以互相转换，最终结果一致。源程序是目标程序产生的基础和前提，目标程序是源程序编译的必然结果；源程序和目标程序具有独立的表现形式，但是目标程序的修改通常依赖于源程序。同一程序的源程序文本和目标程序文本应当视为同一程序。无论是用源程序形式还是目标程序形式体现，都可能得到著作权法保护。

计算机软件包含了计算机程序，并且不局限于计算机程序，还包括与之相关的程序描述和辅助资料。我国将计算机程序文档（软件文档）视为计算机软件的一个组成部分。计算机程序文档与计算机程序不同，计算机程序是用编程语言，如汇编语言、C 语言、Java 语言等编写而成，而计算机程序文档是由自然语言或由形式语言编写而成的。计算机程序文档是指用自然语言或者形式化语言所编写的文字资料和图表，用来描述程序的内容、组成、设计、功能、开发情况、测试结果及使用方法等。计算机程序文档一般以程序设计说明书、流程图、数据流图和用户手册等表现。

我国《计算机软件保护条例》第 6 条规定："本条例对软件著作权的保护不延及开发软件所用的思想、处理过程、操作方法或者数学概念等。"也就是说，软件开发的思想、处理过程、操作方法或者数学概念等与计算机软件分别属于主客观两个范畴。思想是开发软件的设计方案、构思技巧和功能，设计程序所实现的处理过程、操作方法、算法等，表现是完成某项功能的程序。

我国著作权法只保护作品的表达，不保护作品的思想、原理、概念、方法、公式、算法等，因此对计算机软件来说，只有程序的作品性能得到著作权法的保护，而体现其工具性的程序构思、程序技巧等却无法得到保护。实际上计算机程序的技术设计，如软件开发中对软件功能、结构的构思，往往是比程序代码更重要的技术成果，通常体现了软件开发中的主要创造性贡献。

参考答案

（10）D

试题（11）

中国企业 M 与美国公司 L 进行技术合作，合同约定 M 使用一项在有效期内的美国专利，但该项美国专利未在中国和其他国家提出申请。对于 M 销售依照该专利生产的产品，以下叙述正确的是＿＿（11）＿＿。

（11）A. 在中国销售，M 需要向 L 支付专利许可使用费

B. 返销美国，M 不需要向 L 支付专利许可使用费

C. 在其他国家销售，M 需要向 L 支付专利许可使用费

D. 在中国销售，M 不需要向 L 支付专利许可使用费

试题（11）分析

本题考查知识产权知识，涉及专利权的相关概念。知识产权受地域限制，只有在一定地域内知识产权才具有独占性。也就是说，各国依照其本国法律授予的知识产权，只能在其本国领域内受其法律保护，而其他国家对这种权利没有保护的义务，任何人均可在自己的国家内自由使用外国人的知识产品，既无须取得权利人的同意（授权），也不必向权利人支付报酬。例如，中国专利局授予的专利权或中国商标局核准的商标专用权只能在中国领域内受保护，在其他国家则不给予保护。外国人在我国领域外使用中国专利局授权的发明专利不侵犯我国专利权，如美国人在美国使用我国专利局授权的发明专利不侵犯我国专利权。

通过缔结有关知识产权的国际公约或双边互惠协定的形式，某一国家的国民（自然人或法人）的知识产权在其他国家（缔约国）也能取得权益。参加知识产权国际公约的国家（或者签订双边互惠协定的国家）会相互给予成员国国民的知识产权保护。所以，我国公民、法人完成的发明创造要想在外国受保护，必须在外国申请专利。商标要想在外国受保护，必须在外国申请商标注册。著作权虽然自动产生，但它受地域限制，我国法律对外国人的作品并不是都给予保护，只保护共同参加国际条约国家的公民作品。同样，参加公约的其他成员国也按照公约规定，对我国公民和法人的作品给予保护。虽然众多知识产权国际条约等的订立使地域性有时会变得模糊，但地域性的特征不但是知识产权最“古老”的特征，也是最基础的特征之一。目前知识产权的地域性仍然存在，是否授予权利、如何保护权利仍须由各缔约国按照其国内法来决定。

本题涉及的依照该专利生产的产品在中国或其他国家销售，中国 M 企业不需要向美国 L 公司支付这件美国专利的许可使用费。这是因为 L 公司未在中国及其他国家申请该专利，不受中国及其他国家专利法的保护，因此依照该专利生产的产品在中国及其他国家销售，M 企业不需要向 L 公司支付这件专利的许可使用费。如果返销美国，需要向 L 公司支付这件专利的许可使用费。这是因为这件专利已在美国获得批准，因而受到美国专利法的保护，M 企业依照该专利生产的产品要在美国销售，则需要向 L 公司支付这件专利的许可使用费。

参考答案

（11）D

试题（12）

使用<u>（12）</u>DPI 的分辨率扫描一幅 2×4 英寸的照片，可以得到一幅 300×600 像素的图像。

（12）A. 100　　B. 150　　C. 300　　D. 600

试题（12）分析

本题考查多媒体基础知识。我们经常遇到的分辨率有两种，即显示分辨率和图像分辨率。显示分辨率是指显示屏上能够显示出的像素数目。例如，显示分辨率为 1024×768

表示显示屏分成 768 行（垂直分辨率），每行（水平分辨率）显示 1024 个像素，整个显示屏就含有 796 432 个显像点。屏幕能够显示的像素越多，说明显示设备的分辨率越高，显示的图像质量越高。图像分辨率是指组成一幅图像的像素密度，也是用水平和垂直的像素表示，即用每英寸多少点（dpi）表示数字化图像的大小。例如，用 200dpi 来扫描一幅 2×2.5 英寸的彩色照片，那么得到一幅 400×500 个像素点的图像。它实质上是图像数字化的采样间隔，由它确立组成一幅图像的像素数目。对同样大小的一幅图，如果组成该图的图像像素数目越多，则说明图像的分辨率越高，图像看起来就越逼真。相反，图像显得越粗糙。因此，不同的分辨率会造成不同的图像清晰度。

参考答案

（12）B

试题（13）、（14）

计算机数字音乐合成技术主要有__（13）__两种方式，其中使用__（14）__合成的音乐，其音质更好。

（13）A. FM 和 AM　B. AM 和 PM　C. FM 和 PM　D. FM 和 Wave Table

（14）A. FM　B. AM　C. PM　D. Wave Table

试题（13）、（14）分析

本题考查多媒体基础知识。计算机和多媒体系统中的声音，除了数字波形声音之外，还有一类是使用符号表示的，由计算机合成的声音包括语音合成和音乐合成。音乐合成技术主要有调频（FM）音乐合成、波形表（Wave Table）音乐合成两种方式。调频音乐合成是使高频振荡波的频率按调制信号规律变化的一种调制方式。采用不同调制波频率和调制指数就可以方便地合成具有不同频谱分布的波形，再现某些乐器的音色。可以采用这种方法得到具有独特效果的“电子模拟声”，创造出丰富多彩的声音，是真实乐器所不具备的音色。波形表音乐合成是将各种真实乐器所能发出的所有声音（包括各个音域、声调）录制下来，存储为一个波表文件。播放时，根据 MIDI 文件记录的乐曲信息向波表发出指令，从“表格”中逐一找出对应的声音信息，经过合成、加工后回放出来。应用调频音乐合成技术的乐音已经很逼真，波形表音乐合成技术的乐音更真实。目前这两种音乐合成技术都应用于多媒体计算机的音频卡中。

参考答案

（13）D　（14）D

试题（15）

数据流图（DFD）对系统的功能和功能之间的数据流进行建模，其中顶层数据流图描述了系统的__（15）__。

（15）A. 处理过程　B. 输入与输出　C. 数据存储　D. 数据实体

试题（15）分析

本题考查数据流图的基本概念。数据流图从数据传递和加工的角度，以图形的方式

刻画数据流从输入到输出的移动变换过程，其基础是功能分解。对于复杂一些的实际问题，在数据流图中常常出现许多加工，这样看起来不直观，也不易理解，因此用分层的数据流图来建模。按照系统的层次结构进行逐步分解，并以分层的数据流图反映这种结构关系。

在分层的数据流图中，各层数据流图之间应保持“平衡”关系，即输入和输出数据流在各层应该是一致的。

参考答案

（15）B

试题（16）

模块 A 执行几个逻辑上相似的功能，通过参数确定该模块完成哪一个功能，则该模块具有__（16）__内聚。

（16）A. 顺序　　B. 过程　　C. 逻辑　　D. 功能

试题（16）分析

本题考查软件设计的相关内容。模块独立性是创建良好设计的一个重要原则，一般采用模块间的耦合和模块的内聚两个准则进行度量。内聚是指模块内部各元素之间联系的紧密程度，内聚度越高，则模块的独立性越好。内聚性一般有以下几种：

① 偶然内聚：指一个模块内的各个处理元素之间没有任何联系。

② 逻辑内聚：指模块内执行几个逻辑上相似的功能，通过参数确定该模块完成哪一个功能。

③ 时间内聚：把需要同时执行的动作组合在一起形成的模块。

④ 通信内聚：指模块内所有处理元素都在同一个数据结构上操作，或者指各处理使用相同的输入数据或者产生相同的输出数据。

⑤ 顺序内聚：指一个模块中各个处理元素都密切相关于同一功能且必须顺序执行，前一个功能元素的输出就是下一个功能元素的输入。

⑥ 功能内聚：是最强的内聚，指模块内所有元素共同完成一个功能，缺一不可。

参考答案

（16）C

试题（17）

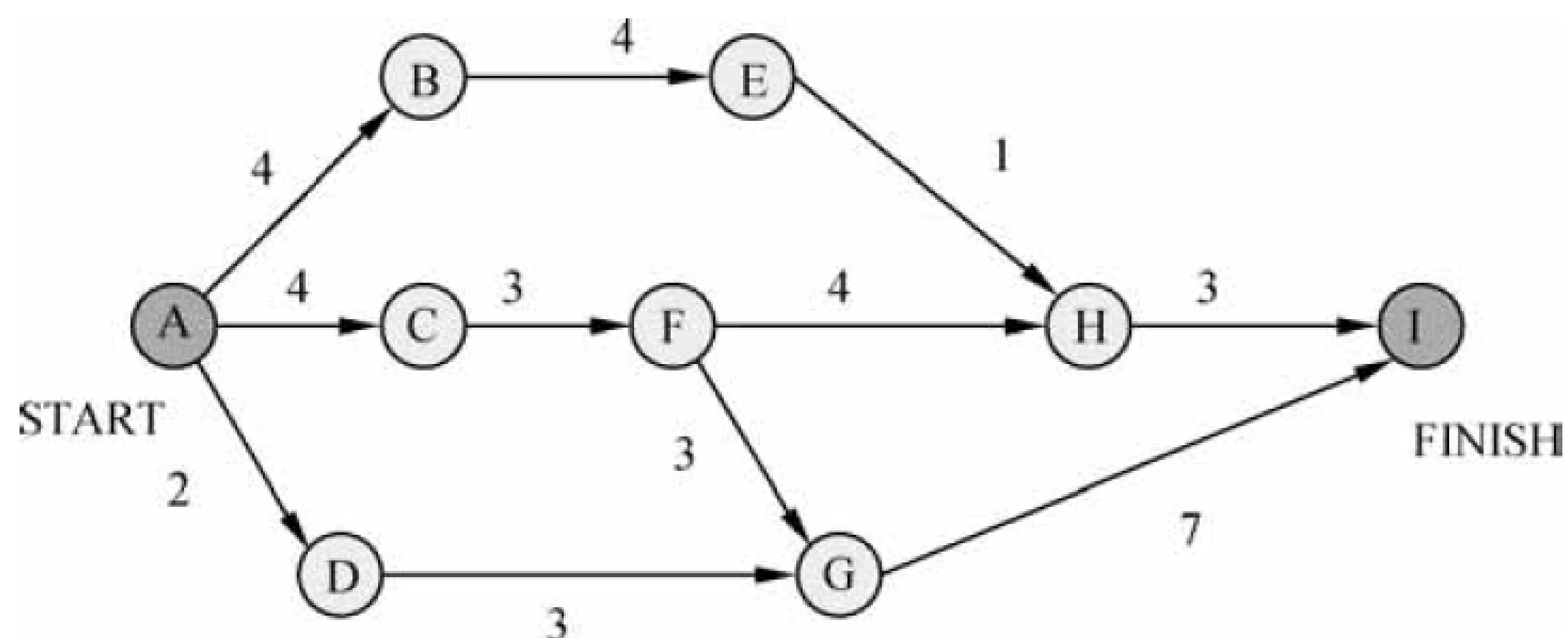
下图是一个软件项目的活动图，其中顶点表示项目里程碑，连接顶点的边表示活动，边上的值表示完成活动所需要的时间，则__（17）__在关键路径上。

（17）A. B　　B. C　　C. D　　D. H

试题（17）分析

本题考查项目管理及工具技术。

根据关键路径法，计算出关键路径为 A—C—F—G—I，关键路径长度为 17。因此里程碑 C 在关键路径上，而里程碑 B、D 和 H 不在关键路径上。



参考答案

（17）B

试题（18）

<u>（18）</u>最不适于采用无主程序员组的开发人员组织形式。

（18）A. 项目开发人数少（如 3～4 人）的项目

B. 采用新技术的项目

C. 大规模项目

D. 确定性较小的项目

试题（18）分析

本题考查项目管理的人员管理。程序设计小组的组织形式一般有主程序员组、无主程序员组和层次式程序员组。其中无主程序员组中的成员之间相互平等，工作目标和决策都由全体成员民主讨论。对于项目规模较小、开发人员少、采用新技术和确定性较小的项目比较合适，而对大规模项目不适宜采用。

参考答案

（18）C

试题（19）

若软件项目组对风险采用主动的控制方法，则<u>（19）</u>是最好的风险控制策略。

（19）A. 风险避免　　B. 风险监控

C. 风险消除　　D. 风险管理及意外事件计划

试题（19）分析

本题考查项目管理的风险管理。风险控制的目的是辅助项目组建立处理风险的策略。有效的策略必须考虑以下三个问题，即风险避免、风险监控和风险管理及意外事件计划，而其中风险避免是最好的风险控制策略。

参考答案

（19）A

试题（20）

对于逻辑表达式“x and y or not z”，and、or、not 分别是逻辑与、或、非运算，优先级从高到低为 not、and、or，and、or 为左结合，not 为右结合，若进行短路计算，则<u>（20）</u>。

（20）A. x 为真时，整个表达式的值即为真，不需要计算 y 和 z 的值
B. x 为假时，整个表达式的值即为假，不需要计算 y 和 z 的值
C. x 为真时，根据 y 的值决定是否需要计算 z 的值
D. x 为假时，根据 y 的值决定是否需要计算 z 的值

试题（20）分析

本题考查程序语言基础知识。

对逻辑表达式可以进行短路计算，其依据是：a and b 的含义是 a 和 b 同时为“真”，则 a and b 为“真”，因此，若 a 为“假”，则无论 b 的值为“真”或“假”，a and b 必然为“假”；a or b 的含义是 a 和 b 同时为“假”，则 a or b 为“假”，因此，若 a 为“真”，则无论 b 的值为“真”或“假”，a or b 必然为“真”。

在优先级和结合性规定下，对逻辑表达式“x and y or not z”求值时，应先计算“x and y”的值，若为“假”，才去计算“not z”的值。因此，若 x 的值为“假”，则“x and y”的值为“假”，需要计算“not z”来确定表达式的值而不管 y 是“真”是“假”。当 x 的值为“真”，则需要计算 y 的值：若 y 的值为“真”，则整个表达式的值为“真”（从而不需再计算“not z”）；若 y 的值为“假”，则需要计算“not z”来确定表达式的值。

参考答案

（20）C

试题（21）

对于二维数组 a[1..N, 1..N]中的一个元素 a[i,j]（1≤i,j≤N），存储在 a[i,j]之前的元素个数__（21）__。

（21）A. 与按行存储或按列存储方式无关
B. 在 i=j 时与按行存储或按列存储方式无关
C. 在按行存储方式下比按列存储方式下要多
D. 在按行存储方式下比按列存储方式下要少

试题（21）分析

本题考查数组元素的存储。

二维数组 a[1..N, 1..N]的元素布局如下：

a[1,1]	a[1,2]	…	a[1,j]	…	a[1,N]
a[2,1]	a[2,2]	…	a[2,j]	…	a[2,N]
⋮	⋮		⋮		⋮
a[i,1]	a[i,2]	…	a[i,j]	…	a[i,N]
⋮	⋮		⋮		⋮
a[N,1]	a[N,2]	…	a[N,j]	…	a[N,N]

在按行存储方式下，a[i,j]之前的元素个数为(i–1)*N+j–1；在按列存储方式下，a[i,j]

之前的元素个数为(j–1)*N+i–1。若 i=j，则 a[i,j]是主对角线上的元素，显然(i–1)*N+j–1 与(j–1)*N+i–1 相等。若 i<j，则 a[i,j]是上三角区域的元素；若 i>j，则 a[i,j]是下三角区域的元素，这两种情况下，存储在 a[i,j]之前的元素个数分别为(i–1)*N+j–1 和(j–1)*N+i–1，其大小关系依赖于 i 和 j 的具体取值。

参考答案

（21）B

试题（22）

算术表达式 x–(y+c)*8 的后缀式是__（22）__（–、+、*表示算术的减、加、乘运算，运算符的优先级和结合性遵循惯例）。

（22）A. x y c 8 – + *　　　　B. x y – c +8 *

C. x y c 8 * + –　　　　D. x y c +8 * –

试题（22）分析

本题考查程序语言基础知识。

后缀表达式（也叫逆波兰式，Reverse Polish notation）是将运算符写在操作数之后的表达式表示方法。

表达式“x–(y+c)*8”的后缀式为“xyc+8*–”。

参考答案

（22）D

试题（23）～（25）

若某企业拥有的总资金数为 15，投资 4 个项目 P1、P2、P3、P4，各项目需要的最大资金数分别是 6、8、8、10，企业资金情况如图 a 所示。P1 新申请 2 个资金，P2 新申请 1 个资金，若企业资金管理处为项目 P1 和 P2 分配新申请的资金，则 P1、P2、P3、P4 尚需的资金数分别为__（23）__；假设 P1 已经还清所有投资款，企业资金使用情况如图 b 所示，那么企业的可用资金数为__（24）__。若在图 b 所示的情况下，企业资金管理处为 P2、P3、P4 各分配资金数 2、2、3，则分配后 P2、P3、P4 已用资金数分别为__（25）__。

项目	最大资金	已用资金	尚需资金
P1	6	2	4
P2	8	3	5
P3	8	2	6
P4	10	3	7

图 a

项目	最大资金	已用资金	尚需资金
P1	—	—	—
P2	8	3	5
P3	8	2	6
P4	10	3	7

图 b

（23）A. 1、3、6、7，可用资金数为 0，故资金周转状态是不安全的

B. 2、5、6、7，可用资金数为 1，故资金周转状态是不安全的

C. 2、4、6、7，可用资金数为 2，故资金周转状态是安全的
D. 3、3、6、7，可用资金数为 2，故资金周转状态是安全的

（24）A. 4　　B. 5　　C. 6　　D. 7

（25）A. 3、2、3，尚需资金数分别为 5、6、7，故资金周转状态是安全的
B. 5、4、6，尚需资金数分别为 3、4、4，故资金周转状态是安全的
C. 3、2、3，尚需资金数分别为 5、6、7，故资金周转状态是不安全的
D. 5、4、6，尚需资金数分别为 3、4、4，故资金周转状态是不安全的

试题（23）～（25）分析

本题考查操作系统进程管理方面的基础知识。

在图 a 的情况下，项目 P1 申请 2 个资金，P2 申请 1 个资金，则企业资金管理处分配资金后项目 P1、P2、P3、P4 已用的资金数分别为 4、4、2、3，可用资金数为 2，故尚需的资金数分别为 2、4、6、7。由于可用资金数为 2，能保证项目 P1 完成。假定项目 P1 完成释放资源后，可用资金数为 6，能保证项目 P2 或 P3 完成。同理，项目 P2 完成释放资源后，可用资金数为 10，能保证项目 P3 或 P4 完成，故资金周转状态是安全的。

对于图 b，因为企业的总资金数是 15，企业资金管理处为项目 P2、P3、P4 已分配资金数为 3、2、3，故可用资金数为 7。

在图 b 的情况下，企业资金管理处为项目 P2、P3、P4 已分配资金数为 3、2、3，若企业资金管理处又为项目 P2、P3、P4 分配资金数为 2、2、3，则企业分配后项目 P2、P3、P4 已用资金数分别为 5、4、6，可用资金为 0，尚需资金数分别为 3、4、4，故资金周转状态是不安全的。

参考答案

（23）C　　（24）D　　（25）D

试题（26）～（28）

假设一台按字节编址的 16 位计算机系统，采用虚拟页式存储管理方案，页面的大小为 2K，且系统中没有使用快表（或联想存储器）。某用户程序如图 a 所示，该程序的页面变换表如图 b 所示，表中状态位等于 1 和 0 分别表示页面在内存或不在内存。

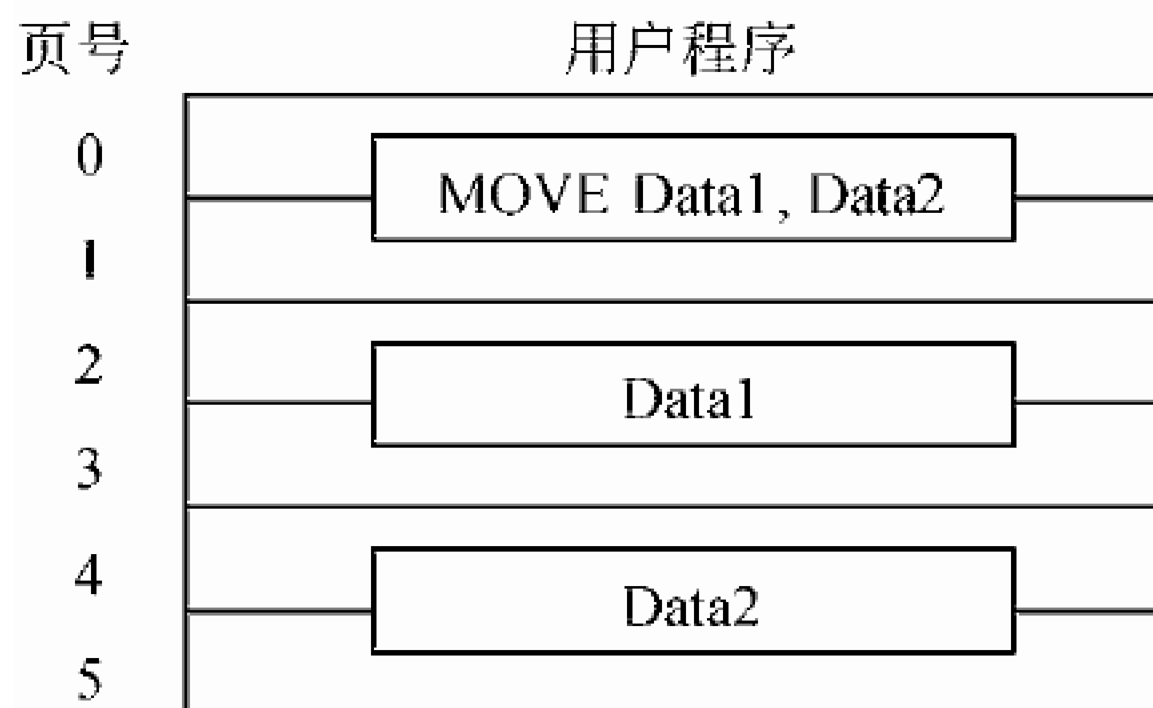


图 a

页面变换表

页号	状态
0	1
1	0
2	0
3	0
4	0
5	0

图 b

图 a 中 MOVE Data1, Data2 是一个 4 字节的指令，Data1 和 Data2 表示该指令的两个 32 位操作数。假设 MOVE 指令存放在 2047 地址开始的内存单元中，Data1 存放在 6143 地址开始的内存单元中，Data2 存放在 10239 地址开始的内存单元中，那么执行 MOVE 指令将产生__（26）__次缺页中断，其中：取指令产生__（27）__次缺页中断，取 Data1 和 Data2 操作数分别产生__（28）__次缺页中断。

（26）A. 3　　B. 4　　C. 5　　D. 6

（27）A. 0　　B. 1　　C. 2　　D. 3

（28）A. 1、1　　B. 1、2　　C. 2、2　　D. 2、3

试题（26）～（28）分析

本题考查操作系统中系统内存管理方面的知识。从题图中可见，程序的 MOVE 指令跨两个页面，且源地址 Data1 和目标地址 Data2 所涉及的区域也跨两个页面的页内地址，根据题意，1、2、3、4 和 5 号页面不在内存，系统取“MOVE Data1, Data2”的指令时，由于该指令跨越两个页面 0、1，查页面变换表可以发现 1 号页面不在内存，故需要产生一次缺页中断；取地址为 Data1 的操作数时，由于该操作数不在内存且跨两个页面 2、3，需要将 2、3 页面装入内存，所以产生两次缺页中断；同理，取地址为 Data2 的操作数时，由于该操作数不在内存且跨两个页面 4、5，需要将 4、5 页面装入内存，所以产生两次缺页中断，共产生 5 次缺页中断。

参考答案

（26）C　　（27）B　　（28）C

试题（29）

软件开发的增量模型__（29）__。

（29）A. 最适用于需求被清晰定义的情况

B. 是一种能够快速构造可运行产品的好方法

C. 最适合于大规模团队开发的项目

D. 是一种不适用于商业产品的创新模型

试题（29）分析

本题考查软件开发过程模型。增量模型是一种阶段化的软件开发过程模型。在该过程模型中，客户提出系统需求，并指出哪些需求是最重要的。开发团队把软件产品作为一系列的增量构件来设计、编码、集成和测试。每个构件由多个相互作用的模块构成，并且能完成特定的功能。其优点包括：能在较短时间内向用户提交可完成一些有用的工作产品；逐步增加产品的功能，使用户有较充裕的时间学习和适应新产品；项目失败的风险较低；优先级最高的服务首先交付，然后依次将其他构件集成进来，这意味着最重要的服务将接受最多的测试。因此增量模式是一种能够快速构造可运行产品的方法，也适用于今天竞争激烈，需要快速发布产品的市场环境。

参考答案

（29）B

试题（30）

假设某软件公司与客户签订合同开发一个软件系统，系统的功能有较清晰的定义，且客户对交付时间有严格要求，则该系统的开发最适宜采用__（30）__。

（30）A. 瀑布模型　B. 原型模型　C. V 模型　D. 螺旋模型

试题（30）分析

本题考查软件过程模型。软件过程是软件生存周期中的一系列相关活动，即用于开发和维护软件及相关产品的一系列活动。瀑布模型从一种非常高层的角度描述了软件开发过程中进行的活动，并且提出了要求开发人员经过的事件序列。该模型适用于项目开始时需求已确定的情况。V 模型是瀑布模型的变种，它说明测试活动是如何与分析和设计相联系的。原型模型允许开发人员快速地构造整个系统或系统的一部分以理解或澄清问题。原型的用途是获知用户的真正需求，因此原型模型可以有效地引发系统需求。螺旋模型把开发活动和风险管理结合起来，以将风险减到最小并控制风险。本题中系统功能有较清晰定义意味着需求较确定，且对交付时间有严格要求，因此最适宜用瀑布模型。

参考答案

（30）A

试题（31）

某企业由于外部市场环境和管理需求的变化对现有软件系统提出新的需求，则对该软件系统进行的维护属于__（31）__维护。

（31）A. 正确性　B. 完善性　C. 适应性　D. 预防性

试题（31）分析

本题考查软件维护技术。

在软件开发完成交付用户使用后，就进入软件运行/维护阶段。软件维护活动根据其内容可以分为 4 种类型：

① 正确性维护。为了识别和纠正软件错误，改正软件性能上的缺陷，排除实施中的误使用，应进行的诊断和改正错误的过程。

② 适应性维护。由于信息技术飞速发展，软件运行的外部环境或数据环境可能发生变化，为了使软件适应这种变化而修改软件的过程。

③ 完善性维护。在软件使用过程中，用户往往会对软件提出新的功能与性能要求，为了满足这些要求，需要修改或再开发软件，以扩充软件功能、增强软件性能、改进加工效率、提高软件的可维护性而进行的维护活动。

④ 预防性维护。为了提高软件的可维护性和可靠性等，为以后进一步改进软件打下良好基础而进行的维护工作。

参考答案

（31）C

试题（32）

McCall 软件质量模型从软件产品的运行、修正和转移三个方面确定了 11 个质量特性，其中__（32）__不属于产品运行方面的质量特性。

（32）A. 正确性　　B. 可靠性　　C. 效率　　D. 灵活性

试题（32）分析

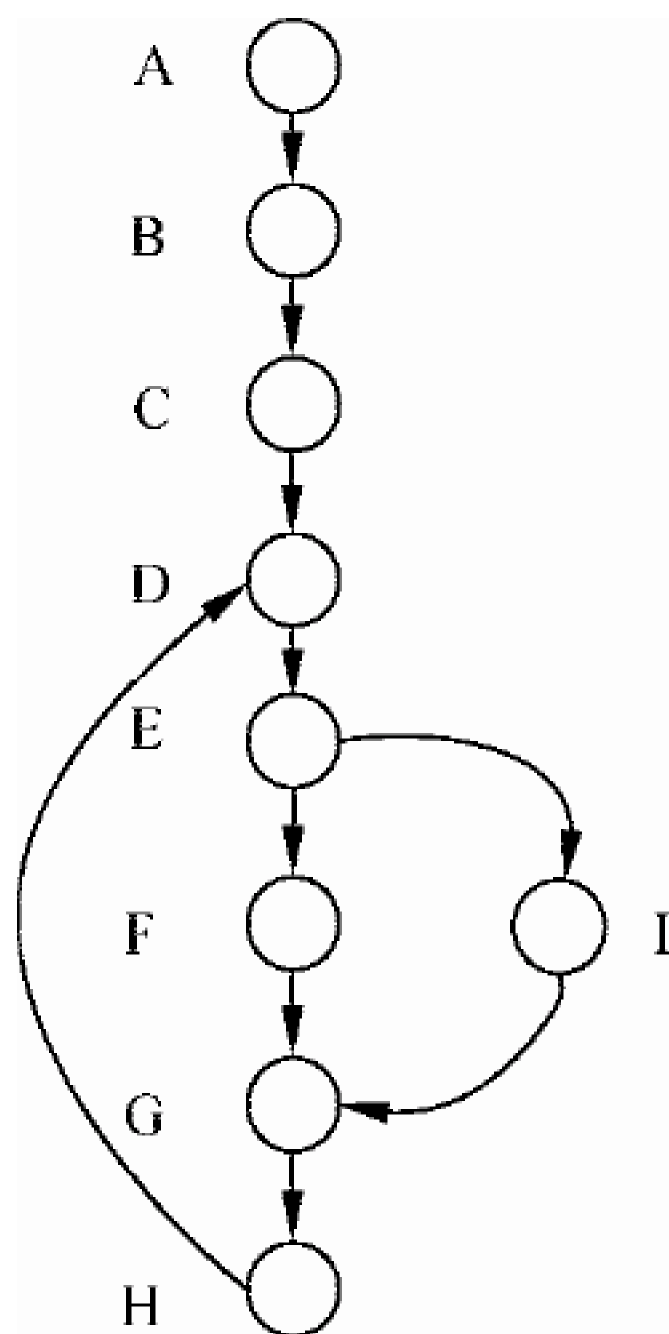
本题考查软件质量的相关知识。McCall 软件质量模型从软件产品的运行、修正和转移三个方面确定了 11 个质量特性。其中产品运行方面包括正确性、可靠性、易使用性、效率和完整性；产品修正方面包括可维护性、灵活性和可测试性；产品转移方面包括可移植性、复用性和互用性。

参考答案

（32）D

试题（33）

采用 McCabe 度量法计算下列程序图的环路复杂性为__（33）__。



（33）A. 2　　B. 3　　C. 4　　D. 5

试题（33）分析

本题考查软件复杂性度量。McCabe 度量法计算程序的环路复杂性为 $V(G) = m - n + 2p$，其中 V(G)是有向图 G 中的环路数，m 是图 G 中弧的个数，n 是图 G 中顶点的个数，p 为图 G 中的强连通分量数。上图中，弧的个数为 10，顶点的个数为 9，p=1，因此有 $V(G) = m - n + 2p = 10 - 9 + 2 = 3$。

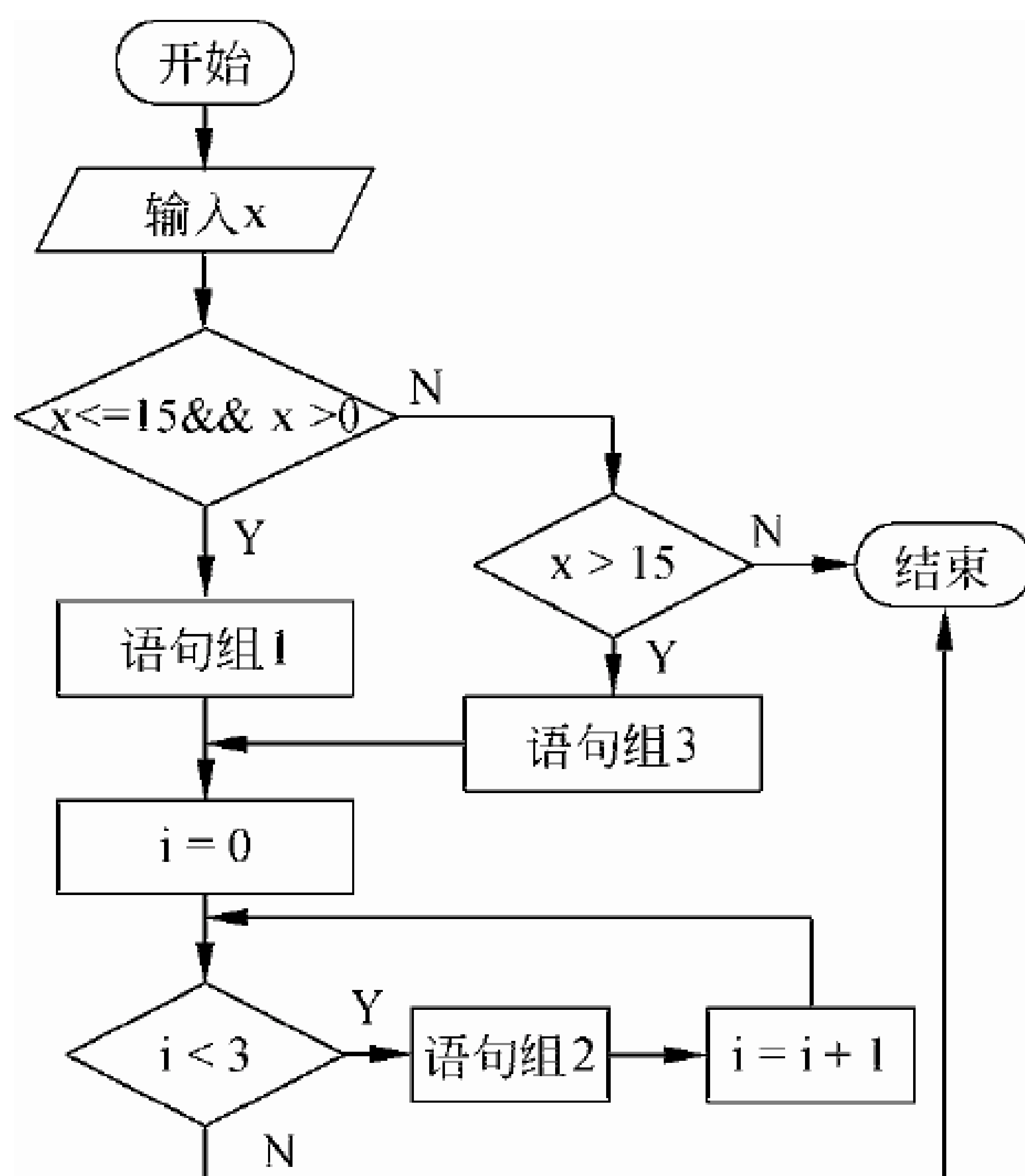
参考答案

（33）B

试题（34）、（35）

在白盒测试法中，__（34）__是最弱的覆盖准则。下图至少需要__（35）__个测试用例

才可以完成路径覆盖，语句组 2 不对变量 i 进行操作。



（34）A. 语句　　B. 条件　　C. 判定　　D. 路径

（35）A. 1　　B. 2　　C. 3　　D. 4

试题（34）、（35）分析

本题考查软件测试的基本概念。

白盒测试也称为结构测试，根据程序的内部结构和逻辑来设计测试用例，对程序的路径和过程进行测试，检查是否满足设计的需要。在白盒测试中，语句覆盖是指选择足够的测试用例，使被测程序中每条语句至少执行一次。它对程序执行逻辑的覆盖很低，因此一般认为是很弱的逻辑覆盖。判定覆盖是指设计足够的测试用例，使得被测程序中每个判定表达式至少获得一次“真”值和“假”值。条件覆盖是指设计足够的测试用例，使得每一个判定语句中每个逻辑条件的各种可能的值至少满足一次。路径覆盖是指覆盖被测程序中所有可能的路径。在这些覆盖技术中，从弱到强依次为语句覆盖、判定覆盖、条件覆盖和路径覆盖。在上图中，要完成路径覆盖，至少需要 3 个测试用例才可以，如测试用例（0）、（8）和（16）即可完成路径覆盖，测试用例格式为（x 的值）。

参考答案

（34）A　　（35）C

试题（36）

根据 ISO/IEC 9126 软件质量模型中对软件质量特性的定义，可维护性质量特性的__（36）__子特性是指与为确认经修改软件所需努力有关的软件属性。

（36）A. 易测试性　　B. 易分析性

C. 稳定性　　D. 易改变性

试题（36）分析

本题考查软件质量特性的基础知识。根据 ISO/IEC 9126 软件质量模型的定义，可维护性质量特性包含易分析性、易改变性、稳定性和易测试性 4 个子特性。其中易分析性是指为诊断缺陷或失效原因，或为判定待修改的部分所需努力有关的软件属性；易改变性是指与进行修改、排错或适应环境变换所需努力有关的软件属性；稳定性是指与修改造成未预料效果的风险有关的软件属性；易测试性是指为确认经修改软件所需努力有关的软件属性。

参考答案

（36）A

试题（37）

面向对象技术中，组合关系表示__（37）__。

（37）A. 包与其中模型元素的关系　　B. 用例之间的一种关系
　　C. 类与其对象的关系　　D. 整体与其部分之间的一种关系

试题（37）分析

本题考查面向对象的基本知识。

在面向对象技术中，包用于将关系紧密的模型元素组织在一起，提供一个命名空间，以提供访问控制。用例之间有继承、包含和扩展关系。类是在对象之上的抽象，对象是类的具体化，对定义好的类的属性的不同赋值就可以得到该类的对象实例。组合关系表示整体与其部分之间的一种关系。

参考答案

（37）D

试题（38）

以下关于封装在软件复用中所充当的角色的叙述，正确的是__（38）__。

（38）A. 封装使得其他开发人员不需要知道一个软件组件内部如何工作
　　B. 封装使得软件组件更有效地工作
　　C. 封装使得软件开发人员不需要编制开发文档
　　D. 封装使得软件组件开发更加容易

试题（38）分析

本题考查面向对象的基本知识。

封装是一种信息隐藏技术，其目的是使对象（组件）的使用者和生产者分离，也就是使其他开发人员无须了解所要使用的软件组件内部的工作机制，只需知道如何使用组件，即组件提供的功能及其接口。

参考答案

（38）A

试题（39）

在有些程序设计语言中，过程调用和响应调用需执行的代码的绑定直到运行时才进行，这种绑定称为__（39）__。

（39）A. 静态绑定　　B. 动态绑定　　C. 过载绑定　　D. 强制绑定

试题（39）分析

本题考查面向对象的基本知识。

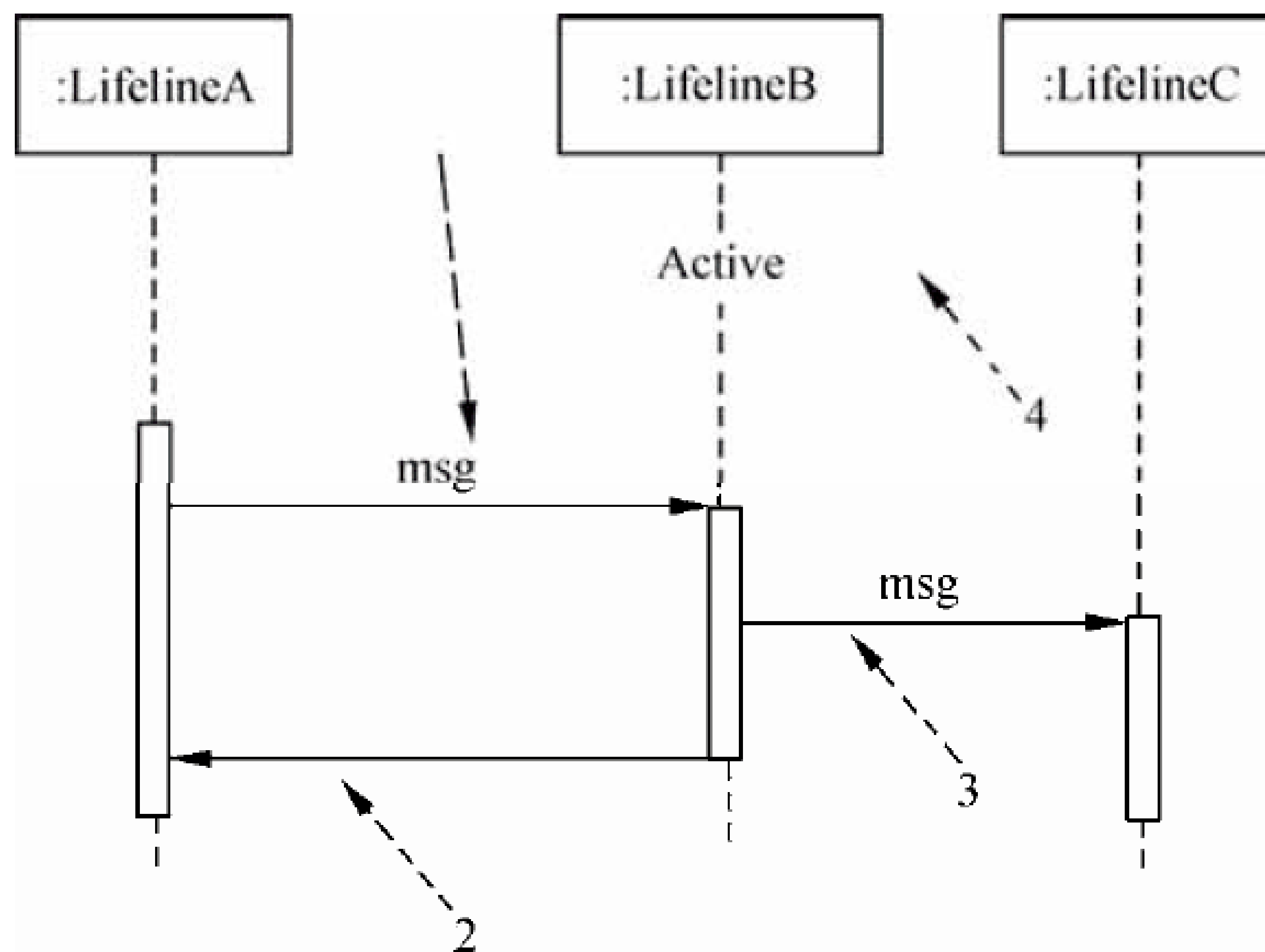
在面向对象系统中，绑定是一个把过程调用和响应调用需要执行的代码加以结合的过程。在有些程序设计语言中，绑定是在编译时进行的，叫做静态绑定。在有些程序设计语言中，绑定则是在运行时进行的，即一个给定的过程调用和响应调用需执行的代码的结合直到调用发生时才进行。

参考答案

（39）B

试题（40）～（43）

UML 序列图是一种交互图，描述了系统中对象之间传递消息的时间次序。其中，异步消息与同步消息不同，__（40）__。下图中__（41）__表示一条同步消息，__（42）__表示一条异步消息，__（43）__表示一条返回消息。



（40）A. 异步消息并不引起调用者终止执行而等待控制权的返回
　　B. 异步消息和阻塞调用有相同的效果
　　C. 异步消息是同步消息的响应
　　D. 异步消息和同步消息一样等待返回消息

（41）A. 1　　B. 2　　C. 3　　D. 4

（42）A. 1　　B. 2　　C. 3　　D. 4

（43）A. 1　　B. 2　　C. 3　　D. 4

试题（40）～（43）分析

本题考查统一建模语言（UML）的基本知识。

UML 2.0 中提供了多种图形，序列图是场景的图形化表示，描述了以时间顺序组织的对象之间的交互活动。其中消息定义了交互中生命线之间的特定交互，有同步消息、异步消息和返回消息三类。同步消息指进行阻塞调用，调用者中止执行，等待控制权返回，需要等待返回消息；而异步消息的调用者发出消息后继续执行，不引起调用者阻塞，也不等待返回消息。消息由名称进行标识，还描述出消息的发出者和接收者。异步消息由空心箭头表示，如上图中 3 所示，同步消息用实心三角箭头表示，如上图中 1 所示，返回消息。

参考答案

（40）A　　（41）A　　（42）C　　（43）B

试题（44）

设计模式根据目的进行分类，可以分为创建型、结构型和行为型三种。其中结构型模式用于处理类和对象的组合。__（44）__模式是一种结构型模式。

（44）A. 适配器（Adapter）　　B. 命令（Command）
　　C. 生成器（Builder）　　D. 状态（State）

试题（44）分析

本题考查设计模式的基本知识。

每一个设计模式描述了一个在我们周围不断重复发生的问题，以及该问题的解决方案的核心，使该方案能够重用而不必做重复劳动。

设计模式根据目的进行分类，可以分为创建型、结构型和行为型三种。其中创建型模式与对象的创建有关；结构型模式用于处理类和对象的组合；行为型模式描述类或对象怎样交互和怎样分配职责。

适配器（Adapter）模式是一种结构型模式；命令（Command）模式和状态（State）模式是行为型模式；生成器（Builder）模式是一种创建型模式。

参考答案

（44）A

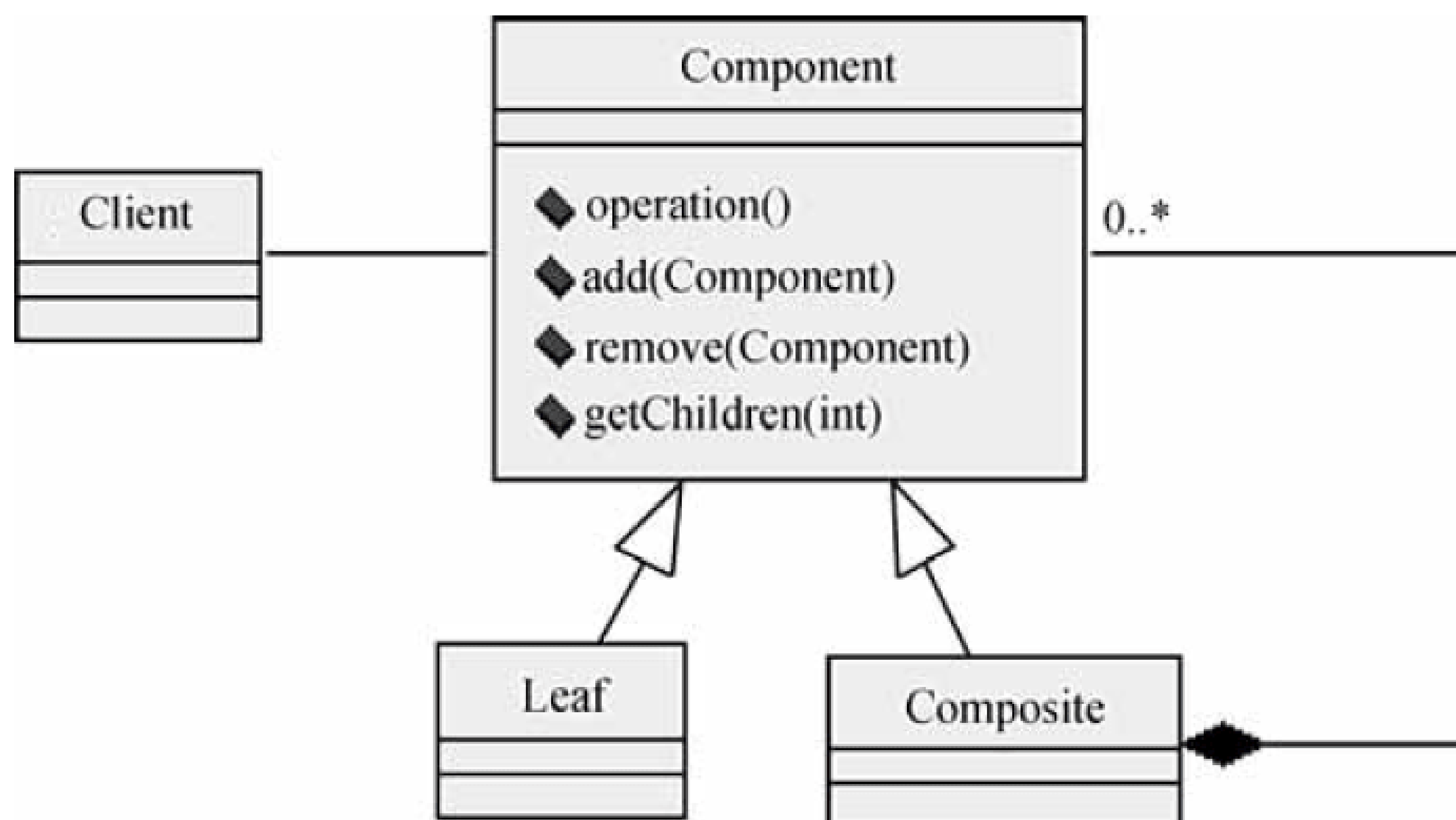
试题（45）～（47）

设计模式中的__（45）__模式将对象组合成树形结构以表示“部分-整体”的层次结构，使得客户对单个对象和组合对象的使用具有一致性。下图为该模式的类图，其中，__（46）__定义有子部件的那些部件的行为；组合部件的对象由__（47）__通过 Component 提供的接口操作。

（45）A. 代理（Proxy）　　B. 桥接器（Bridge）
　　C. 组合（Composite）　　D. 装饰器（Decorator）

（46）A. Client　　B. Component　　C. Leaf　　D. Composite

（47）A. Client　　B. Component　　C. Leaf　　D. Composite



试题（45）～（47）分析

本题考查设计模式的基本知识。

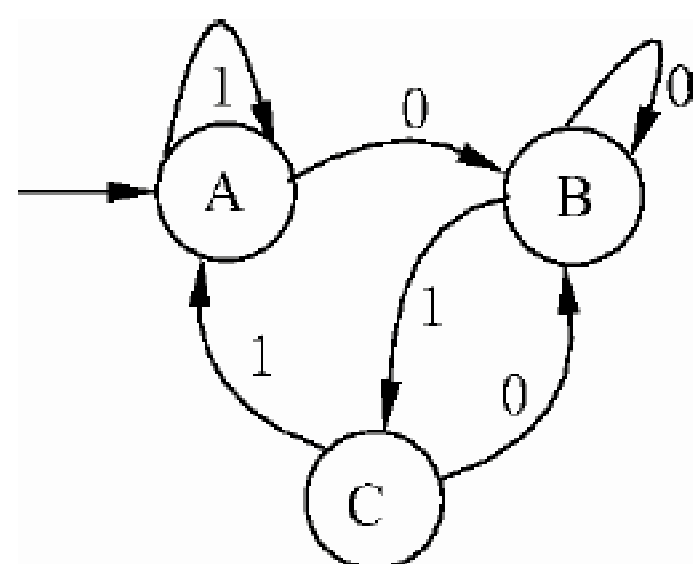
代理（Proxy）模式是为其他对象提供一种代理以控制对这个对象的访问，使得只有在确实需要这个对象时才对其进行创建和初始化。桥接器（Bridge）模式将对象的抽象和其实现分离，从而可以独立地改变它们，当一个抽象可能有多个实现时，抽象类定义对该抽象的接口，而具体的子类则用不同方式加以实现。组合（Composite）模式描述了如何将对象组合成树形结构以构造一个层次结构来表示“部分-整体”，使得客户对单个对象和组合对象的使用具有一致性，这一结构由两种类型的对象所对应的类构成，使得可以组合基元对象以及其他的组合对象，从而形成任意复杂的结构。上图中的 Composite 定义有子部件的那些部件的行为；组合部件的对象由 Client 通过 Component 提供的接口操作。装饰器（Decorator）模式则描述动态地给一个对象添加一些额外的职责。

参考答案

（45）C （46）D （47）A

试题（48）

下图所示为一个有限自动机（其中，A 是初态、C 是终态），该自动机所识别的字符串的特点是__（48）__。



（48）A. 必须以 11 结尾的 0、1 串 B. 必须以 00 结尾的 0、1 串

C. 必须以 01 结尾的 0、1 串 D. 必须以 10 结尾的 0、1 串

试题（48）分析

本题考查程序语言处理基础知识。

从有限自动机的初态到终态的路径上的标记形成其可识别的字符串。

对于题中的自动机，从 A 出发到达 C 结束的所有路径中必然包含 BC 这条弧（标记为 1），同时到达 B 的弧上都标记了 0，所以其识别的字符串必须以 01 结尾。

参考答案

（48）C

试题（49）

E-R 模型向关系模型转换时，三个实体之间多对多的联系 m:n:p 应该转换为一个独立的关系模式，且该关系模式的关键字由__（49）__组成。

（49）A. 多对多联系的属性　　　　B. 三个实体的关键字

C. 任意一个实体的关键字　　　　D. 任意两个实体的关键字

试题（49）分析

本题考查数据库设计方面的基础知识。

E-R 模型向关系模型转换时，两个以上实体之间多对多的联系应该转换为一个独立的关系模式，且该关系模式的关键字由这些实体的关键字组成。

参考答案

（49）B

试题（50）

函数（过程）调用时，常采用传值与传地址两种方式在实参与形参间传递信息。以下叙述中，正确的是__（50）__。

（50）A. 在传值方式下，将形参的值传给实参，因此，形参必须是常量或变量

B. 在传值方式下，将实参的值传给形参，因此，实参必须是常量或变量

C. 在传地址方式下，将形参的地址传给实参，因此，形参必须有地址

D. 在传地址方式下，将实参的地址传给形参，因此，实参必须有地址

试题（50）分析

本题考查程序语言处理基础知识。

一个函数被调用时，可能需要接收从外部传入的数据信息，传值调用与引用调用（传地址）是函数调用时常采用的信息传递方式。传值调用是将实参的值传给被调用函数的形参，因此实参可以是常量、变量、表达式或函数调用，而引用调用的实质是将实参的地址传给被调用函数的形参，因此实参必须具有地址。

参考答案

（50）D

试题（51）

编译和解释是实现高级程序设计语言翻译的两种基本形式。以下关于编译与解释的叙述中，正确的是__（51）__。

（51）A. 在解释方式下，对源程序不进行词法分析和语法分析，直接进行语义分析

B. 在解释方式下，无须进行词法、语法和语义分析，而是直接产生源程序的目标代码

C. 在编译方式下，必须进行词法、语法和语义分析，然后再产生源程序的目标代码

D. 在编译方式下，必须先形成源程序的中间代码，然后再产生与机器对应的目标代码

试题（51）分析

本题考查程序语言处理基础知识。

程序的翻译通常有两种基本方式：一种是编译方式，另一种是解释方式。

在编译方式下，首先将源程序翻译为等价的目标程序，源程序的翻译和目标程序的运行是完全独立的两个阶段；而解释方式下，对源程序的翻译和运行是结合在一起进行的，并不生成目标代码。

编译过程基本上可以划分为词法分析、语法分析、语义分析、中间代码生成、代码优化和目标代码生成等几个阶段，其中，中间代码生成和代码优化不是必须的。在词法、语法、语义分析方面，编译方式和解释方式没有区别。

参考答案

（51）C

试题（52）、（53）

若对关系 R(A,B,C,D)进行 $\pi_{1,3}(R)$ 运算，则该关系运算与__（52）__等价，表示__（53）__。

（52）A. $\pi_{A=1,C=3}(R)$　B. $\pi_{A=1 \wedge C=3}(R)$　C. $\pi_{A,C}(R)$　D. $\pi_{A=1 \vee C=3}(R)$

（53）A. 属性 A 和 C 的值分别等于 1 和 3 的元组为结果集

B. 属性 A 和 C 的值分别等于 1 和 3 的两列为结果集

C. 对 R 关系进行 A=1、C=3 的投影运算

D. 对 R 关系进行属性 A 和 C 的投影运算

试题（52）、（53）分析

本题考查关系代数运算方面的基础知识。

投影运算π是向关系的垂直方向进行运算，其含义为在关系 R 中选择出若干属性列组成新的关系，记作：$\pi_{A_{i1},A_{i2},\cdots,A_{ii}}(R)=\left\{ t\left[A_{i1},A_{i2},\cdots,A_{in}\right] \mid t\in R\right\}$。本试题中，关系 R（A，B，C，D）共有 4 个属性，属性 A、B、C 和 D 分别位于第 1 列、第 2 列、第 3 列和第 4 列，$\pi_{1,3}(R)$ 相当于在关系 R 的第 1 列和第 3 列上进行投影运算，即在关系 R 的属性 A 和 C 上进行投影运算，故 $\pi_{1,3}(R)$ 与 $\pi_{A,C}(R)$ 是等价的。

参考答案

（52）C　（53）D

试题（54）～（56）

某销售公司数据库的零件关系 P（零件号，零件名称，供应商，供应商所在地，库

存量），函数依赖集 F={零件号→零件名称，（零件号，供应商）→库存量，供应商→供应商所在地}。零件关系模式 P 属于＿（54）＿。

查询各种零件的平均库存量、最多库存量与最少库存量之间差值的 SQL 语句如下：

```
SELECT 零件号，零件名称，____(55)____
FROM P
____(56)____;
```

（54）A. 1NF　　B. 2NF　　C. 3NF　　D. 4NF

（55）A. AVG(库存量) AS 平均库存量, MAX(库存量)−MIN(库存量) AS 差值
　　B. 平均库存量 AS AVG(库存量), 差值 AS MAX(库存量)−MIN(库存量)
　　C. AVG 库存量 AS 平均库存量, MAX 库存量−MIN 库存量 AS 差值
　　D. 平均库存量 AS AVG 库存量, 差值 AS MAX 库存量−MIN 库存量

（56）A. ORDER BY 供应商　　B. ORDER BY 零件号
　　C. GROUP BY 供应商　　D. GROUP BY 零件号

试题（54）～（56）分析

本题考查关系数据库及 SQL 方面的基础知识。

根据题意，零件 P 关系中的（零件号，供应商）可决定零件 P 关系的所有属性，所以零件 P 关系的主键为（零件号，供应商）；又因为，根据题意（零件号，供应商）→零件名称，而零件号→零件名称，供应商→供应商所在地，可以得出零件名称和供应商所在地都部分依赖于码，所以该关系模式属于 1NF。

查询各种零件的平均库存量、最高库存量与最低库存量之间差距时，首先需要在结果列中的空（55）处填写“AVG（库存量）AS 平均库存量，MAX(库存量)−MIN(库存量) AS 差值”。其次必须用分组语句按零件号分组，故空（56）应填写“GROUP BY 零件号”。

参考答案

（54）A　（55）A　（56）D

试题（57）

对于一个长度大于 1 且不存在重复元素的序列，令其所有元素依次通过一个初始为空的队列后，再通过一个初始为空的栈。设队列和栈的容量都足够大，一个序列通过队列（栈）的含义是序列的每个元素都入队列（栈）且出队列（栈）一次且仅一次。对于该序列在上述队列和栈上的操作，正确的叙述是＿（57）＿。

（57）A. 出队序列和出栈序列一定相同
　　B. 出队序列和出栈序列一定互为逆序
　　C. 入队序列与出队序列一定相同，入栈序列与出栈序列不一定相同
　　D. 入栈序列与出栈序列一定互为逆序，入队序列与出队序列不一定互为逆序

试题（57）分析

本题考查数据结构基础知识。

栈和队列是两种常用的数据结构。栈的特点是后进先出，队列的特点是先进先出。因此，入队序列与出队序列一定相同。在入栈序列一定的情况下，由于元素的出栈时机不同，会形成不同的出栈序列，入栈序列与出栈序列可以相同，也可以不同。

参考答案

（57）C

试题（58）

在字符串的 KMP 模式匹配算法中，需要求解模式串 p 的 next 函数值，其定义如下所示。若模式串 p 为“aaabaaa”，则其 next 函数值为＿（58）＿。

$$next[j]=\begin{cases}0 & j=1\\ \max\{k \mid 1<k<j,'p_1p_2\cdots p_{k-1}'='p_{j-k+1}p_{j-k+2}\cdots p_{j-1}'\} & \\ 1 & \text{其他情况}\end{cases}$$

（58）A. 0123123　　B. 0123210　　C. 0123432　　D. 0123456

试题（58）分析

本题考查字符串的模式匹配运算。

KMP 模式匹配算法是对基本模式匹配算法的改进，其改进之处在于：每当匹配过程中出现相比较的字符不相等时，不需要回溯主串的字符位置指针，而是利用已经得到的“部分匹配”结果将模式串向右“滑动”尽可能远的距离，再继续进行比较。

在 KMP 算法中，依据模式串的 next 函数值实现子串的滑动。若令 next[j]=k，则 next[j] 表示当模式串中的 p_j 与主串中相应字符不相等时，令模式串的 p_k 与主串的相应字符进行比较。

根据 next 的定义，模式串“aaabaaa”的 next 函数值为 0123123。

参考答案

（58）A

试题（59）

若 n_2、n_1、n_0 分别表示一个二叉树中度为 2、度为 1 和叶子结点的数目（结点的度定义为结点的子树数目），则对于任何一个非空的二叉树，＿（59）＿。

（59）A. n_2 一定大于 n_1　　B. n_1 一定大于 n_0

C. n_2 一定大于 n_0　　D. n_0 一定大于 n_2

试题（59）分析

本题考查数据结构中二叉树的基础知识。

对任何一棵二叉树，若其终端节点数为 n_0，度为 2 的节点数为 n_2，则 $n_0=n_2+1$。

证明如下：

设一棵二叉树上叶结点数为 n_0，单分支结点数为 n_1，双分支结点数为 n_2，则总结点

数$=n_0+n_1+n_2$。

在一棵二叉树中，所有结点的分支数（即度数）应等于单分支结点数加上双分支结点数的 2 倍，即总的分支数$=n_1+2n_2$。

由于二叉树中除根结点以外，每个结点都有唯一的一个分支指向它，因此二叉树中：总的分支数=总结点数–1。因此，$n_1+2n_2=n_0+n_1+n_2-1$，即 $n_0=n_2+1$。

参考答案

（59）D

试题（60）

从存储空间的利用率角度来看，以下关于数据结构中图的存储的叙述，正确的是__（60）__。

（60）A. 有向图适合采用邻接矩阵存储，无向图适合采用邻接表存储

B. 无向图适合采用邻接矩阵存储，有向图适合采用邻接表存储

C. 完全图适合采用邻接矩阵存储

D. 完全图适合采用邻接表存储

试题（60）分析

本题考查数据结构中图结构基础知识。

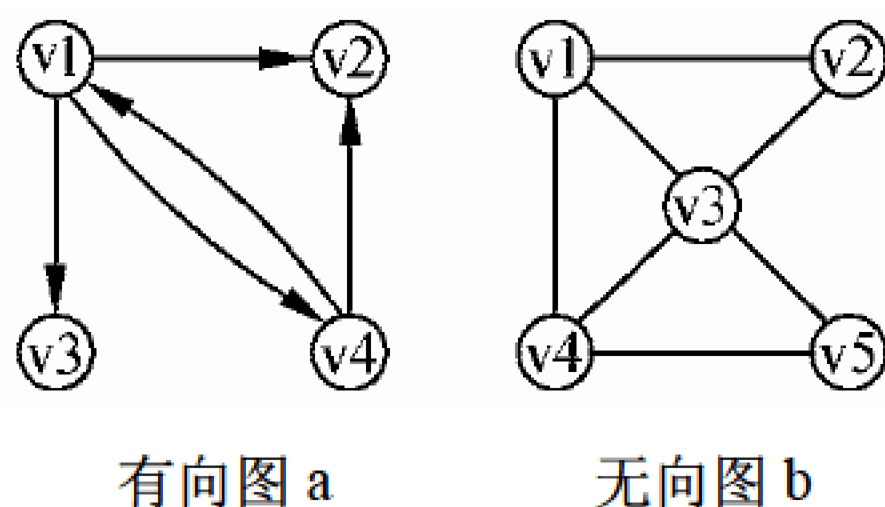
图的基本存储结构有邻接矩阵表示法和邻接链表表示法。

图的邻接矩阵表示利用一个矩阵来表示图中顶点之间的关系。对于具有 n 个顶点的图 G=(V，E)，其邻接矩阵是一个 n 阶方阵，且满足：

$$A[i][j]=\begin{cases}1\text{若}(v_i,v_j)\text{或}<v_i,v_j>\text{是 E 中的}\\0\text{若}(v_i,v_j)\text{或}<v_i,v_j>\text{不是 E 中的}\end{cases}$$

邻接表存储是指为图的每个顶点建立一个单链表，第 i 个单链表中的结点表示依附于顶点 v_i 的边（对于有向图是以 v_i 为尾的弧）。

例如，下面分别为一个有向图 a 和一个无向图 b。

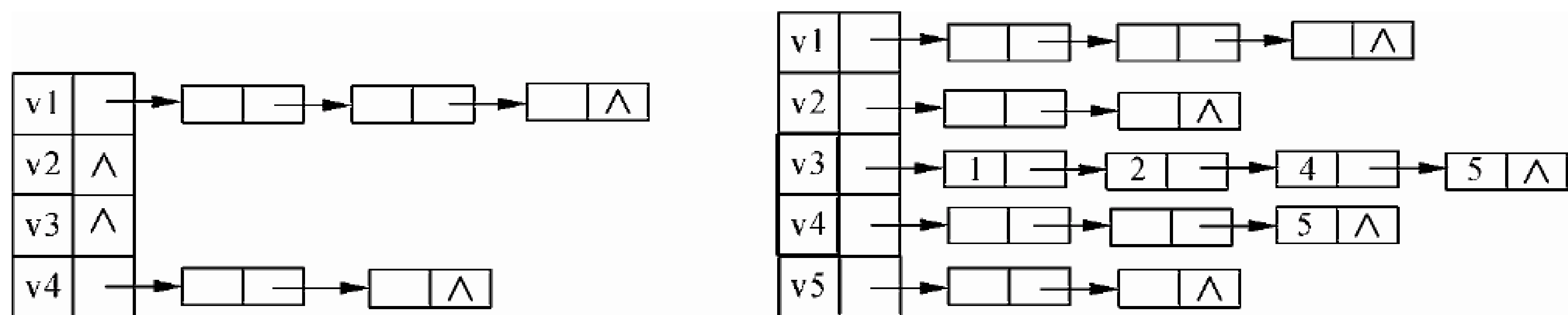


有向图 a　　无向图 b

有向图 a 和无向图 b 的邻接矩阵如下所示。

$$A=\begin{bmatrix}0&1&1&1\\0&0&0&0\\0&0&0&0\\1&1&0&0\end{bmatrix}\qquad B=\begin{bmatrix}0&1&1&1&0\\1&0&1&0&0\\1&1&0&1&1\\1&0&1&0&1\\0&0&1&1&0\end{bmatrix}$$

有向图 a 和无向图 b 的邻接表如下图所示。



图中的顶点数决定了邻接矩阵的阶和邻接表中的单链表数目，无论是对有向图还是无向图，边数的多少决定了单链表中的结点数，而不影响邻接矩阵的规模，因此完全图适合采用邻接矩阵存储。

参考答案

（60）C

试题（61）

递增序列 A（$a_1,a_2,\cdots,a_n$）和 B（$b_1,b_2,\cdots,b_n$）的元素互不相同，若需将它们合并为一个长度为 2n 的递增序列，则当最终的排列结果为__（61）__时，归并过程中元素的比较次数最多。

（61）A. $a_1,a_2,\cdots,a_n,b_1,b_2,\cdots,b_n$

B. $b_1,b_2,\cdots,b_n,a_1,a_2,\cdots,a_n$

C. $a_1,b_1,a_2,b_2,\cdots,a_i,b_i,\cdots,a_n,b_n$

D. $a_1,a_2,\cdots,a_{i/2},b_1,b_2,\cdots,b_{i/2},a_{i/2+1},a_{i/2+2},\cdots,a_n,b_{i/2+1},b_{i/2+2},\cdots,b_n$

试题（61）分析

本题考查归并排序算法。

归并的过程是：取序列 A 的一个元素 a_i 和序列 B 的一个元素 b_j，若 $a_i>b_j$，则输出 b_j，接下来令 a_i 与 b_{j+1} 比较，否则输出 a_i；接下来令 a_{i+1} 与 b_j 比较，重复以上过程直至将所有元素输出。

对于最终排列为 $a_1,a_2,\cdots,a_n,b_1,b_2,\cdots,b_n$ 的情况，归并过程中进行了 n 次比较，分别是 $a_1<b_1$， $a_2<b_1$， …， $a_n<b_1$， 最后依次输出 $b_1,b_2,\cdots,b_n$。

对于最终排列为 $b_1,b_2,\cdots,b_n,a_1,a_2,\cdots,a_n$ 的情况，归并过程中进行了 n 次比较，分别是 $b_1<a_1$， $b_2<a_1$， …， $b_n<a_1$， 最后依次输出 $a_1,a_2,\cdots,a_n$。

对于最终排列为 $a_1,b_1,a_2,b_2,\cdots,a_i,b_i,\cdots,a_n,b_n$ 的情况，归并过程中进行了 2n–1 次比较，分别是 $a_1<b_1$， $b_1<a_2$， $a_2<b_2$， $b_2<a_3$ …， $a_n<b_n$。

若最终排列为 $a_1,a_2,\cdots,a_{i/2},b_1,b_2,\cdots,b_{i/2},a_{i/2+1},a_{i/2+2},\cdots,a_n,b_{i/2+1},b_{i/2+2},\cdots,b_n$，则在归并过程中，分别是 $a_1,a_2,\cdots,a_{i/2}$ 与 b_1 各进行一次比较，共 i/2 次；然后是 $b_1,b_2,\cdots,b_{i/2}$ 分别与 $a_{i/2+1}$ 各进行一次比较，共 i/2 次；接下来是 $a_{i/2+1},a_{i/2+2},\cdots,a_n$ 分别与 $b_{i/2+1}$ 各进行一次比较，共（n–i/2）次，合计比较次数为 i/2+i/2+n–i/2=n+i/2。

参考答案

（61）C

试题（62）

以下关于渐进符号的表示中，不正确的是__（62）__。

（62）A. $n^2=\Theta(n^2)$　　B. $n^2=O(n^2)$　　C. $n^2=O(n)$　　D. $n^2=O(n^3)$

试题（62）分析

本题考查算法分析技术的相关知识。几个算法时间复杂度符号的定义分别如下：

O 记号：给出一个函数的渐进上界。给定一个函数 g(n)，O(g(n))表示为一个函数集合的{f(n)：存在正常数 c 和 n_0，使得对所有的 $n \geqslant n_0$，有 $0 \leqslant f(n) \leqslant cg(n)$}。

Ω 记号：给出一个函数的渐进下界。给定一个函数 g(n)，Ω(g(n))表示为一个函数集合{f(n)：存在正常数 c 和 n_0，使得对所有的 $n \geqslant n_0$，有 $0 \leqslant cg(n) \leqslant f(n)$ }。

Θ 记号：给出一个函数的渐进上界和下界，即渐进确界。给定一个函数 g(n)，Θ(g(n))表示为一个函数集合{f(n)：存在正常数 c_1、c_2 和 n_0，使得对所有的 $n \geqslant n_0$，有 $0 \leqslant c_1 g(n) \leqslant f(n) \leqslant c_2 g(n)$}。

根据定义，可知 C 不正确。

参考答案

（62）C

试题（63）、（64）

某货车运输公司有一个中央仓库和 n 个运输目的地，每天要从中央仓库将货物运输到所有的运输目的地，到达每个运输目的地一次且仅一次，最后回到中央仓库。在两个地点 i 和 j 之间运输货物存在费用 c_{ij}。为求解旅行费用总和最小的运输路径，设计如下算法：首先选择离中央仓库最近的运输目的地 1，然后选择离运输目的地 1 最近的运输目的地 2，……，每次在未访问过的运输目的地中选择离当前运输目的地最近的运输目的地，最后回到中央仓库。则该算法采用了__（63）__算法设计策略，其时间复杂度为__（64）__。

（63）A. 分治　　B. 动态规划　　C. 贪心　　D. 回溯

（64）A. $\Theta(n^2)$　　B. $\Theta(n)$　　C. $\Theta(n\lg n)$　　D. $\Theta(1)$

试题（63）、（64）分析

本题考查算法设计策略。

由于每次选择下一个要访问的城市时都是基于与当前最近的城市来进行，是一种贪心的选择策略，因此采用的是贪心策略。而货车从中央仓库出发，第一个要到达的目的地是在 n 个目的地中选择一个，第二个要到达的目的地是在 n–1 个目的地中选择一个，……，第 n 个要到达的目的地是在 1 个目的地中选择一个，因此时间复杂度为 $n+(n-1)+\cdots+1=n*(n-1)/2=\Theta(n^2)$。

参考答案

（63）C　　（64）A

试题（65）

现要对 n 个实数（仅包含正实数和负实数）组成的数组 A 进行重新排列，使得其中

所有的负实数都位于正实数之前。求解该问题的算法的伪代码如下所示，则该算法的时间和空间复杂度分别为__（65）__。

```
i = 0; j = n-1;
while i<j do
        while A[i] < 0 do
            i = i+1;
        while A[j] > 0 do
            j = j-1;
        if i < j do
```

交换 A[i]和 A[j];

（65）A. $\Theta(n)$ 和 $\Theta(n)$　　B. $\Theta(1)$ 和 $\Theta(n)$

C. $\Theta(n)$ 和 $\Theta(1)$　　D. $\Theta(1)$ 和 $\Theta(1)$

试题（65）分析

本题考查算法分析方法。

根据伪代码可知算法的基本思想，从前往后检查元素，若为负数继续向前检查；若遇到正数，则开始从后往前检查元素，若为正数则继续往前检查；若遇到负数则与前面遇到的正数进行交换。重复检查元素，所有元素检查完毕。根据该思想，可知每个元素检查一遍，因此算法的时间复杂度为线性时间，即 $\Theta(n)$。在该过程中，仅需要一个额外的辅助存储空间，以便进行元素的交换，因此空间复杂度为常数，即 $\Theta(1)$。

参考答案

（65）C

试题（66）

以下关于网络中各种交换设备的叙述中，错误的是__（66）__。

（66）A. 以太网交换机根据 MAC 地址进行交换

B. 帧中继交换机只能根据虚电路号 DLCI 进行交换

C. 三层交换机只能根据第三层协议进行交换

D. ATM 交换机根据虚电路标识进行信元交换

试题（66）分析

以太网交换机根据数据链路层 MAC 地址进行帧交换；帧中继网和 ATM 网都是面向连接的通信网，交换机根据预先建立的虚电路标识进行交换。帧中继的虚电路号是 DLCI，进行交换的协议数据单元为“帧”；而 ATM 网的虚电路号为 VPI 和 VCI，进行交换的协议数据单元为“信元”。

三层交换机是指因特网中使用的高档交换机，这种设备把 MAC 交换的高带宽和低延迟优势与网络层分组路由技术结合起来，其工作原理可以概括为：一次路由，多次交换。就是说，当三层交换机第一次收到一个数据包时必须通过路由功能寻找转发端口，

同时记住目标 MAC 地址和源 MAC 地址，以及其他相关信息，当再次收到目标地址和源地址相同的帧时就直接进行交换了，不再调用路由功能。所以三层交换机不但具有路由功能，而且比通常的路由器转发得更快。

参考答案

（66）C

试题（67）

SMTP 传输的邮件报文采用__（67）__格式表示。

（67）A．ASCII　　B．ZIP　　C．PNP　　D．HTML

试题（67）分析

本题考查 SMTP 协议及相关服务。SMTP 传输的邮件报文需采用 ASCII 进行编码。

参考答案

（67）A

试题（68）

网络的可用性是指__（68）__。

（68）A. 网络通信能力的大小　　B. 用户用于网络维修的时间

C. 网络的可靠性　　D. 用户可利用网络时间的百分比

试题（68）分析

可用性是指网络系统、网络元素或网络应用对用户可利用的时间的百分比。有些应用对可用性很敏感，例如飞机订票系统若宕机一小时，就可能减少几十万元的票款；而股票交易系统如果中断运行一分钟，就可能造成几千万元的损失。实际上，可用性是网络元素可靠性的表现，而可靠性是指网络元素在具体条件下完成特定功能的概率。

如果用平均无故障时间（Mean Time Between Failure，MTBF）来度量网络元素的故障率，则可用性 A 可表示为 MTBF 的函数：

$$A=\frac{\text{MTBF}}{\text{MTBF}+\text{MTTR}}$$

其中 MTTR（Mean Time To Repair）为发生失效后的平均维修时间。由于网络系统由许多网络元素组成，因此系统的可靠性不但与各个元素的可靠性有关，而且还与网络元素的组织形式有关。根据可靠性理论，由元素串并联组成的系统的可用性与网络元素的可用性之间的关系如下图所示。从图 a 中可以看出，若两个元素串联，则可用性减少。例如两个 Modem 串联在链路的两端，若单个 Modem 的可用性 A=0.98，并假定链路其他部分的可用性为 1，则整个链路的可用性 A=0.98×0.98=0.9604。从图 b 中可以看出，若两个元素并联，则可用性增加。例如终端通过两条链路连接到主机，若一条链路失效，另外一条链路自动备份。假定单个链路的可用性 A=0.98，则双链路的可用性 A=2×0.98–0.98×0.98=1.96–0.9604=0.9996。

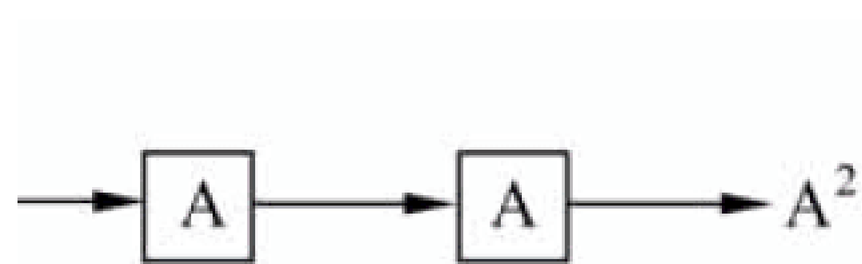


图 a 串联

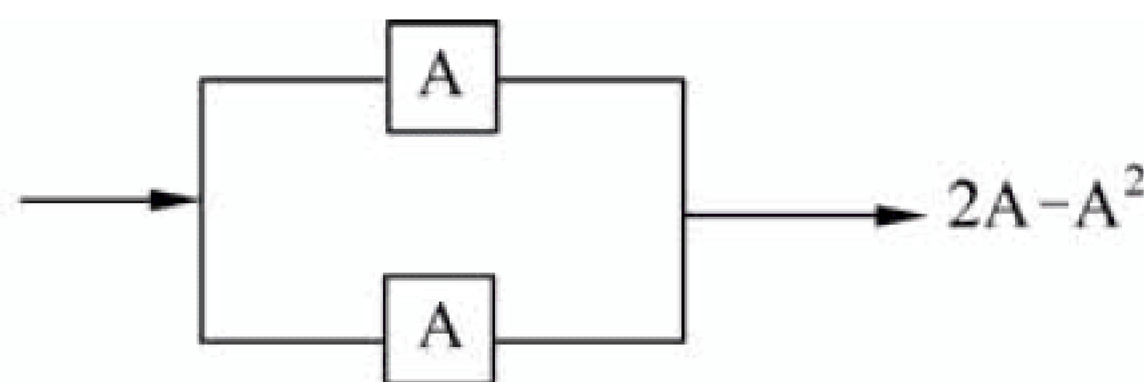


图 b 并联

参考答案

（68）D

试题（69）

建筑物综合布线系统中的园区子系统是指__（69）__。

（69）A. 由终端到信息插座之间的连线系统

B. 楼层接线间到工作区的线缆系统

C. 各楼层设备之间的互连系统

D. 连接各个建筑物的通信系统

试题（69）分析

结构化综合布线系统（Structure Cabling System）是基于现代计算机技术的通信物理平台，集成了语音、数据、图像和视频的传输功能，消除了原有通信线路在传输介质上的差别。

结构化布线系统分为 6 个子系统：工作区子系统、水平子系统、干线子系统、设备间子系统、管理子系统和建筑群子系统。

（1）工作区子系统（Work Location）。

工作区子系统是由终端设备到信息插座的整个区域。一个独立的需要安装终端设备的区域划分为一个工作区。工作区应支持电话、数据终端、计算机、电视机、监视器以及传感器等多种终端设备。

（2）水平布线子系统（Horizontal）。

各个楼层接线间的配线架到工作区信息插座之间所安装的线缆属于水平子系统。水平子系统的作用是将干线子系统线路延伸到用户工作区。

（3）管理子系统（Administration）。

管理子系统设置在楼层的接线间内，由各种交连设备（双绞线跳线架、光纤跳线架）以及集线器和交换机等交换设备组成，交连方式取决于网络拓扑结构和工作区设备的要求。交连设备通过水平布线子系统连接到各个工作区的信息插座，集线器或交换机与交连设备之间通过短线缆互连，这些短线被称为跳线。通过跳线的调整，可以在工作区的信息插座和交换机端口之间进行连接切换。

（4）干线子系统（Backbone）。

干线子系统是建筑物的主干线缆，实现各楼层设备间子系统之间的互连。干线子系统通常由垂直的大对数铜缆或光缆组成，一头端接于设备间的主配线架上，另一头端接在楼层接线间的管理配线架上。

（5）设备间子系统（Equipment）。

建筑物的设备间是网络管理人员值班的场所，设备间子系统由建筑物的进户线、交换设备、电话、计算机、适配器以及保安设施组成，实现中央主配线架与各种不同设备（如 PBX、网络设备和监控设备等）之间的连接。

（6）建筑群子系统（Campus）。

建筑群子系统也叫园区子系统，它是连接各个建筑物的通信系统。大楼之间的布线方法有三种：一种是地下管道敷设方式，管道内敷设的铜缆或光缆应遵循电话管道和入孔的各种规定，安装时至少应预留 1～2 个备用管孔，以备扩充之用。第二种是直埋法，要在同一个沟内埋入通信和监控电缆，并应设立明显的地面标志。最后一种是架空明线，这种方法需要经常维护。

参考答案

（69）D

试题（70）

如果子网 172.6.32.0/20 被划分为子网 172.6.32.0/26，则下面的结论中正确的是<u>（70）</u>。

（70）A. 被划分为 62 个子网　　　　B. 每个子网有 64 个主机地址

　　　C. 被划分为 32 个子网　　　　D. 每个子网有 62 个主机地址

试题（70）分析

子网 172.6.32.0/20 被划分为子网 172.6.32.0/26，网络掩码增加了 6 位，被划分成了 64 个子网，每个子网的主机 ID 部分为 6 位，可以提供主机地址个数为 62。

参考答案

（70）D

试题（71）～（75）

At a basic level, cloud computing is simply a means of delivering IT resources as <u>（71）</u>. Almost all IT resources can be delivered as a cloud service: applications, compute power, storage capacity, networking, programming tools, even communication services and collaboration <u>（72）</u>.

Cloud computing began as large-scale Internet service providers such as Google, Amazon, and others built out their infrastructure. An architecture emerged: massively scaled, <u>（73）</u> distributed system resources, abstracted as virtual IT services and managed as continuously configured, pooled resources. In this architecture, the data is mostly resident on <u>（74）</u> “somewhere on the Internet” and the application runs on both the “cloud servers” and the user’s browser.

Both clouds and grids are built to scale horizontally very efficiently. Both are built to withstand failures of <u>（75）</u> elements or nodes. Both are charged on a per-use basis. But while grids typically process batch jobs, with a defined start and end point, cloud services can

be continuous. What's more, clouds expand the types of resources available — file storage, databases, and Web services — and extend the applicability to Web and enterprise applications.

（71）A. hardware　　B. computers　　C. services　　D. software
（72）A. computers　　B. disks　　C. machine　　D. tools
（73）A. horizontally　　B. vertically　　C. inclined　　D. decreasingly
（74）A. clients　　B. middleware　　C. servers　　D. hard disks
（75）A. entire　　B. individual　　C. general　　D. separate

试题（71）～（75）分析

本题考查对英语资料的阅读理解。

本段英文简要介绍云计算的概念。云计算主要是将资源看作云服务，包括应用程序、计算能力、存储容量、网络、编程工具，以及通信和协作工具。云计算最初由一些大的 Internet 服务提供商构建的基础设施而起步，其架构呈现出大规模、水平分布式系统资源、抽象的 IT 服务、管理持续配置、资源池等特性，数据大多存储于 Internet 上的某个地方的服务器上，应用程序运行于云服务器和用户浏览器中。

云和网格都针对有效的水平可扩展性，避免节点的单点失效对系统的影响，都按使用付费。它们的区别是网格通常是处理一批有明确定义起点和终点的作业，而云服务是可以连续不断的。另外，云扩展了资源的类型，包括文件存储、数据库和 Web 服务等，也将适用性扩展到 Web 和企业应用。

参考答案

（71）C　　（72）D　　（73）A　　（74）C　　（75）B

第 14 章　2012 上半年软件设计师下午试题分析与解答

试题一（共 15 分）

阅读下列说明和图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

【说明】

某学校欲开发图书管理系统，以记录图书馆所藏图书及其借出和归还情况，提供给借阅者借阅图书功能，提供给图书馆管理员管理和定期更新图书表功能。主要功能的具体描述如下：

（1）处理借阅。借阅者要借阅图书时，系统必须对其身份（借阅者 ID）进行检查。通过与教务处维护的学生数据库、人事处维护的职工数据库中的数据进行比对，以验证借阅者 ID 是否合法。若合法，则检查借阅者在逾期未还图书表中是否有逾期未还图书，以及罚金表中的罚金是否超过限额。如果没有逾期未还图书并且罚金未超过限额，则允许借阅图书，更新图书表，并将借阅的图书存入借出图书表。借阅者归还所借图书时，先由图书馆管理员检查图书是否缺失或损坏，若是，则对借阅者处以相应罚金并存入罚金表；然后，检查所还图书是否逾期，若是，执行“处理逾期”操作；最后，更新图书表，删除借出图书表中的相应记录。

（2）维护图书。图书馆管理员查询图书信息；在新进图书时录入图书信息，存入图书表；在图书丢失或损坏严重时，从图书表中删除该图书记录。

（3）处理逾期。系统在每周一统计逾期未还图书，逾期未还的图书按规则计算罚金，并记入罚金表，并给有逾期未还图书的借阅者发送提醒消息。借阅者在借阅和归还图书时，若罚金超过限额，管理员收取罚金，并更新罚金表中的罚金额度。

现采用结构化方法对该图书管理系统进行分析与设计，获得如图 1-1 所示的顶层数据流图和图 1-2 所示的 0 层数据流图。

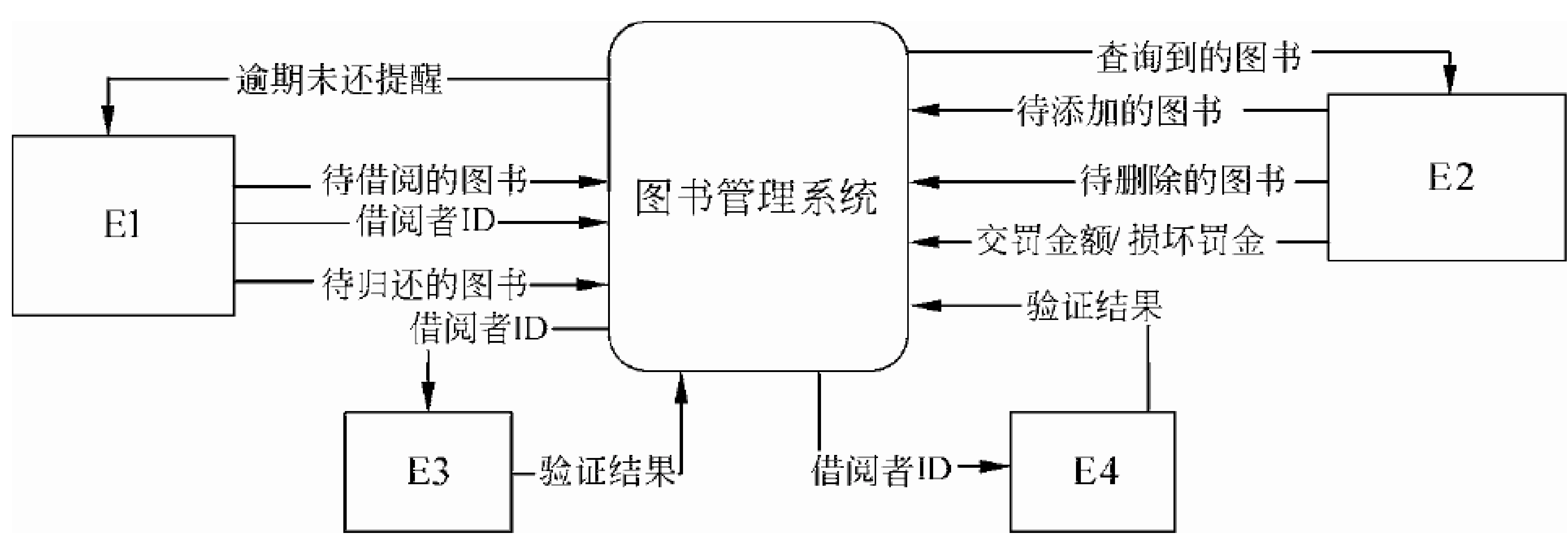


图 1-1　顶层数据流图

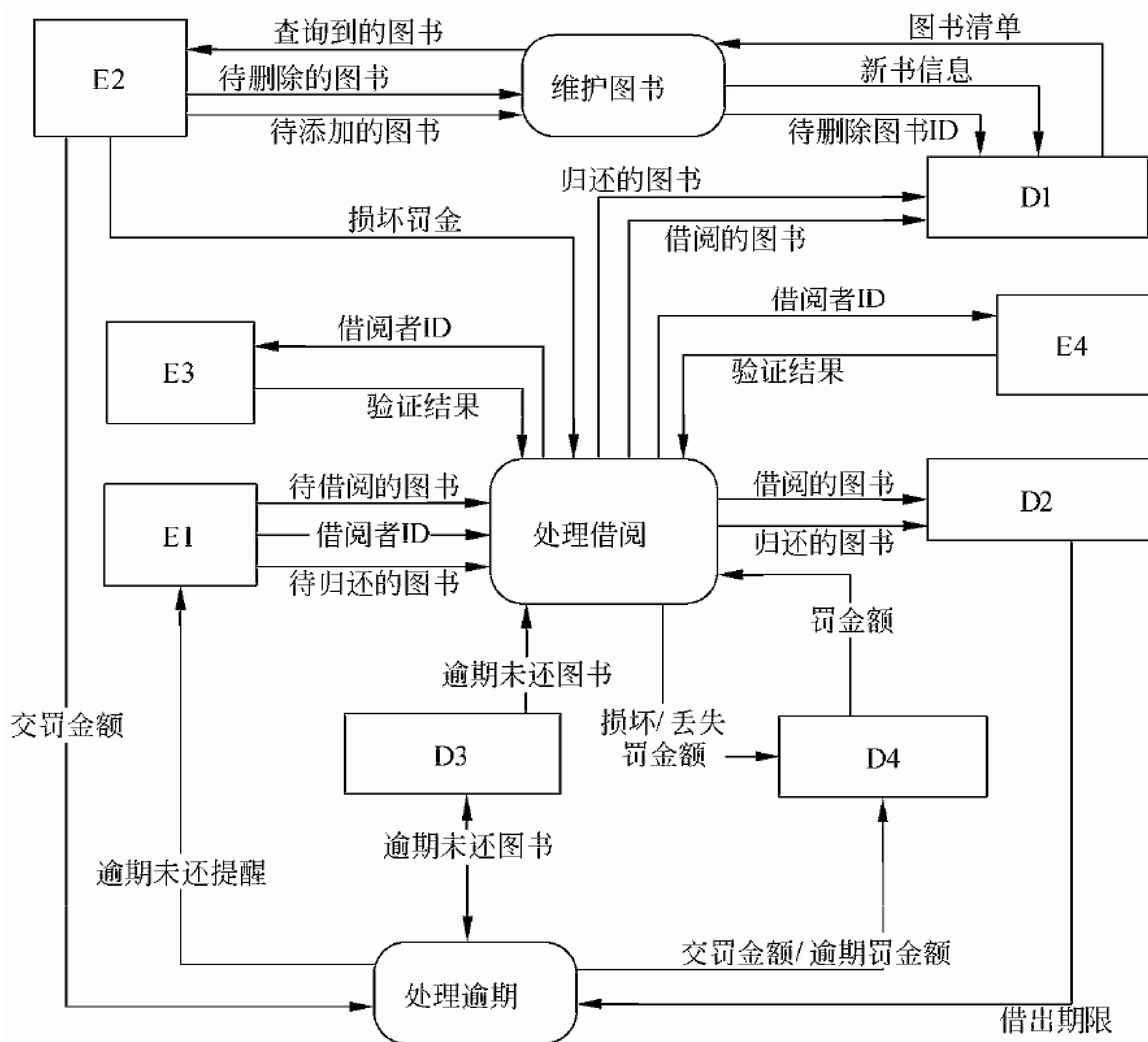


图 1-2　0 层数据流图

【问题 1】（4 分）

使用说明中的词语，给出图 1-1 中的实体 E1～E4 的名称。

【问题 2】（4 分）

使用说明中的词语，给出图 1-2 中的数据存储 D1～D4 的名称。

【问题 3】（5 分）

在 DFD 建模时，需要对有些复杂加工（处理）进行进一步精化，绘制下层数据流图。针对图 1-2 中的加工“处理借阅”，在 1 层数据流图中应分解为哪些加工？（使用说明中的术语）

【问题 4】（2 分）

说明问题 3 中绘制 1 层数据流图时要注意的问题。

试题一分析

本题考查采用结构化方法进行系统分析与设计，主要考查数据流图（DFD）的应用，是比较传统的题目，要求考生细心分析题目中所描述的内容。

DFD 是一种便于用户理解、分析系统数据流程的图形化建模工具，是系统逻辑模型

的重要组成部分。

【问题 1】

本问题考查顶层 DFD。顶层 DFD 一般用来确定系统边界，将待开发系统看作一个加工，图中只有唯一的一个处理和一些外部实体，以及这两者之间的输入输出数据流。题目要求根据描述确定图中的外部实体。分析题目中描述，并结合已经在顶层数据流图中给出的数据流进行分析。从题目的说明中可以看出：和系统的交互者包括图书管理员、借阅者两类人，图书管理员需要维护图书信息、得到查询所得的图书信息，借阅者提供借阅者 ID、借阅与归还的图书。还有通过与教务处维护的学生数据库、人事处维护的职工数据库中的数据进行比对以验证借阅者 ID 是否合法的两个数据库作为外部实体。

对应图 1-1 中数据流和实体的对应关系，可知 E1 为借阅者，E2 为图书管理员，E3 和 E4 为学生数据库和职工数据库。

【问题 2】

本问题考查 0 层 DFD 中数据存储的确定。说明中描述维护图书信息主要存储或者更新图书表；借阅时需要检查逾期未还图书表，确定是否有逾期未还图书以及罚金表中的罚金限额，归还时出现缺失和损坏需要处以罚金并存入罚金表；借阅与归还图书时需要存入借出图书表和更新借出图书表。在处理逾期时需要将罚金记入罚金表，要检查和更新罚金限额。根据描述和图 1-2 中的数据存储的输入输出数据流提示，可知 D1 为图书表，D2 为借出图书表，D3 为逾期未还图书表，D4 为罚金表。

【问题 3】

本问题考查将 0 层 DFD 中的处理进一步精化建模，绘制下层数据流图。从说明中对“处理借阅”的描述和图 1-2 可知，处理借阅需要检查借阅者身份、检查逾期未还图书、检查罚金是否超过限额、借阅图书和归还图书。描述中：检查所还图书是否逾期，若是，执行“处理逾期”操作。这里处理逾期明确说明是一个操作，而且在描述（3）中单独描述，图 1-2 中已经建模为单独一个处理，所以在本问题中仍然不分解。

【问题 4】

本问题考查在绘制下层数据流图时需要注意的问题。问题 3 明确给出是对复杂处理进行进一步精化，绘制下层数据流图，因此需要注意的问题是绘制下层数据流图时要保持父图与子图平衡。父图中某加工的输入输出数据流必须与它的子图的输入输出数据流在数量和名字上相同。如果父图的一个输入（或输出）数据流对应于子图中几个输入（或输出）数据流，而子图中组成这些数据流的数据项全体正好是父图中的这一个数据流，那么它们仍然算是平衡的。

参考答案

【问题 1】

E1：借阅者　　　　E2：图书管理员　　　　E3/E4：学生数据库/职工数据库

注：E3 和 E4 不分顺序，但必须不同。

【问题 2】

D1：图书表 D2：借出图书表 D3：逾期未还图书表 D4：罚金表

【问题 3】

检查借阅者身份或检查借阅者 ID；检查逾期未还图书；检查罚金是否超过限额；借阅图书；归还图书。

【问题 4】

保持父图与子图平衡。父图中某加工的输入输出数据流必须与它的子图的输入输出数据流在数量和名字上相同。如果父图的一个输入（或输出）数据流对应于子图中几个输入（或输出）数据流，而子图中组成这些数据流的数据项全体正好是父图中的这一个数据流，那么它们仍然算是平衡的。

试题二（共 15 分）

阅读下列说明，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某医院拟开发一套住院病人信息管理系统，以方便对住院病人、医生、护士和手术等信息进行管理。

【需求分析】

（1）系统登记每个病人的住院信息，包括：病案号、病人的姓名、性别、地址、身份证号、电话号码、入院时间及病床等信息，每个病床有唯一所属的病区及病房，如表 2-1 所示。其中病案号唯一标识病人本次住院的信息。

表 2-1 住院登记表

病案号	071002286	姓名	张三	性别	男
身份证号	0102196701011234	入院时间	2011-03-03	病床号	052401
病房	0524 室	病房类型	三人间	所属病区	05 II 区

（2）在一个病人的一次住院期间，由一名医生对该病人的病情进行诊断，并填写一份诊断书，如表 2-2 所示。对于需要进行一次或多次手术的病人，系统记录手术名称、手术室、手术日期、手术时间、主刀医生及多名协助医生，每名医生在手术中的责任不同，如表 2-3 所示，其中手术室包含手术室号、楼层、地点和类型等信息。

表 2-2 诊断书

诊断时间：2011 年 03 月

病案号	071002286	姓名	张三	性别	男	医生	李**
诊断							

（3）护士分为两类：病床护士和手术室护士。每个病床护士负责护理一个病区内的所有病人，每个病区由多名护士负责护理。手术室护士负责手术室的护理工作。每个手

术室护士负责多个手术室，每个手术室由多名护士负责，每个护士在手术室中有不同的责任，并由系统记录其责任。

表 2-3　手术安排表

手术名称	***手术	病案号	071002286	姓名	张三	性别	男
手术室	032501	手术日期	2011-03-15	手术时间	8:30~10:30	主刀医生	李**
协助医生	王**(协助)，周**(协助)，刘**(协助)，高**(麻醉)						

【概念模型设计】

根据需求阶段收集的信息，设计的实体联系图（不完整）如图 2-1 所示。

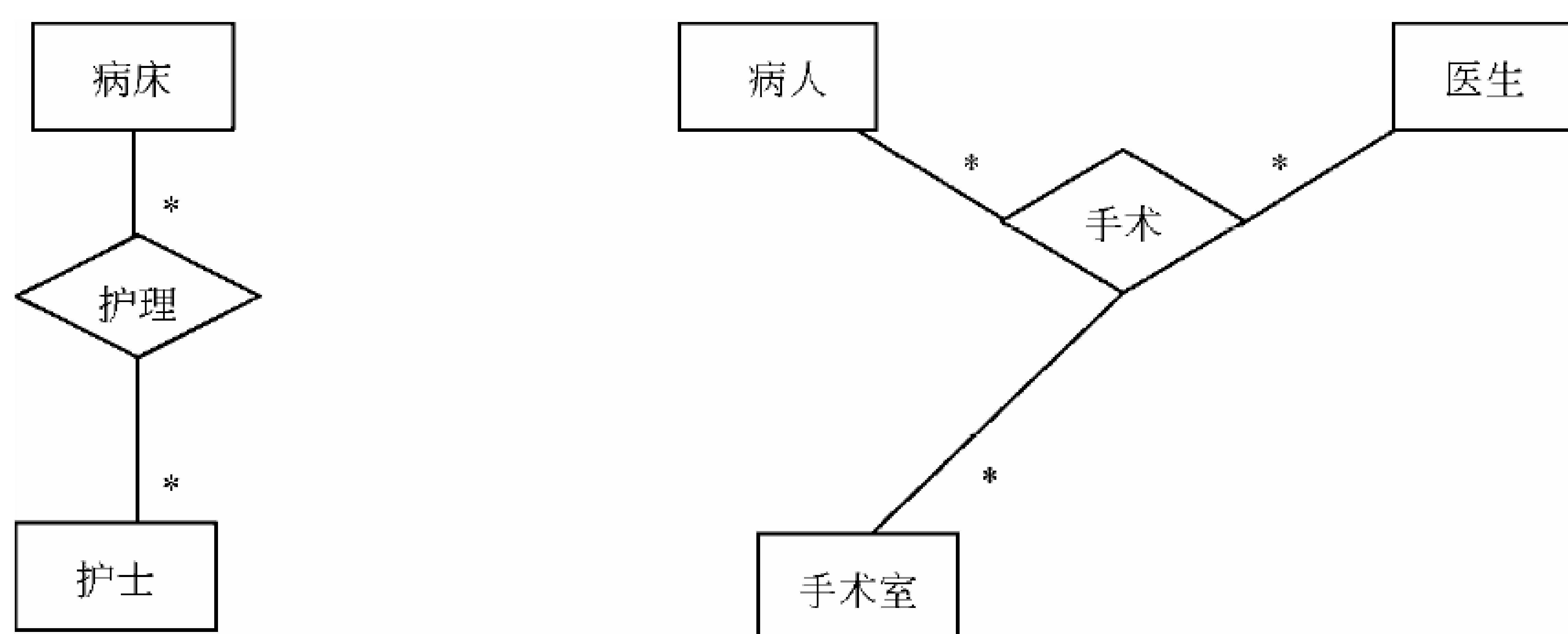


图 2-1　实体联系图

【逻辑结构设计】

根据概念模型设计阶段完成的实体联系图，得出如下关系模式（不完整）：

病床（病床号，病房，病房类型，所属病区）
护士（护士编号，姓名，类型，性别，级别）
病床护士（＿＿＿(1)＿＿＿）
手术室（手术室号，楼层，地点，类型）
手术室护士（＿＿＿(2)＿＿＿）
病人（＿＿＿(3)＿＿＿，姓名，性别，地址，身份证号，电话号码，入院时间）
医生（医生编号，姓名，性别，职称，所属科室）
诊断书（＿＿＿(4)＿＿＿，诊断，诊断时间）
手术安排（病案号，手术室号，手术时间，手术名称）
手术医生安排（＿＿＿(5)＿＿＿，医生责任）

【问题 1】（6 分）

补充图 2-1 中的联系和联系的类型。

【问题 2】（5 分）

根据图 2-1，将逻辑结构设计阶段生成的关系模式中的空（1）～（5）补充完整，并用下划线指出主键。

【问题 3】（4 分）

如果系统还需要记录医生给病人的用药情况，即记录医生给病人所开处方中药品的名称、用量、价格、药品的生产厂家等信息。请根据该要求，对图 2-1 进行修改，画出补充后的实体、实体间联系和联系的类型。

试题二分析

本题考查数据库设计，属于比较传统的题目，考查点也与往年类似。

【问题 1】

本问题考查数据库的概念结构设计，题目要求补充完整实体联系图中的联系和联系的类型。

根据题目的需求描述可知，一名病人在一次住院期间对应一张病床，而一个病床可以有多名病人曾经住过。所以，病床实体和病人实体之间存在“住院”联系，联系的类型为多对一，表示为*:1。

根据题目的需求描述可知，一名病人在一次住院期间，由一名医生做出诊断，并给出一份诊断书。所以，病人实体和医生实体之间存在“诊断”联系，联系的类型为多对多，表示为*:1。

根据题目的需求描述可知，一名病人在一次住院期间可以进行多次手术，一次手术安排在一个手术室，由多名医生参与。所以，病人实体与医生实体和手术室实体三者之间存在“手术”联系，三者之间联系的类型为多对多对多，表示为*:*:*。

根据题目的需求描述可知，一名手术室护士负责多个手术室，每个手术室由多名护士负责。所以，护士实体和手术室实体之间存在“负责”联系，联系的类型为多对多，表示为*:*。

【问题 2】

本问题考查数据库的逻辑结构设计，题目要求补充完整各关系模式，并给出各关系模式的主键。

根据实体联系图和需求描述，每个病床护士负责护理一个病区内的所有病人，每个病区由多名护士负责护理。系统记录每个病床护士所负责护理的病区。所以，对于“病床护士”关系模式需填写的属性为：病区，护士号。

根据实体联系图和需求描述，每个手术室护士负责多个手术室，每个手术室由多名护士负责，每个护士在手术室中有不同的责任。因此，对于“手术室护士”关系模式，需填写的属性为：手术室号，护士号，责任。

根据实体联系图和需求描述，病案号唯一标识病人本次住院的信息。病人的住院信息包括病床信息。所以，对于“病人”关系模式需补充的属性为：病案号，病床号。

根据实体联系图和需求描述，一名病人在一次住院期间，由一名医生做出诊断，并给出一份诊断书。所以，对于“诊断”关系模式需补充的属性为：病案号，医生编号。

根据实体联系图和需求描述，一名病人在一次住院期间，可能需要进行一次或多次手术，每次手术安排在一间手术室，由多名医生(包括主刀医生)参与。所以，对于“手术医生安排”关系模式需补充的属性为：病案号，手术室号，手术时间，医生编号。

病床护士关系模式的主键：病区，护士号

手术室护士关系模式的主键：手术室号，护士号

病人关系模式的主键：病案号

诊断书关系模式的主键：病案号

采购订单关系模式的主键：订单编码

手术医生安排关系模式的主键：病案号，手术室号，手术时间，医生编号码

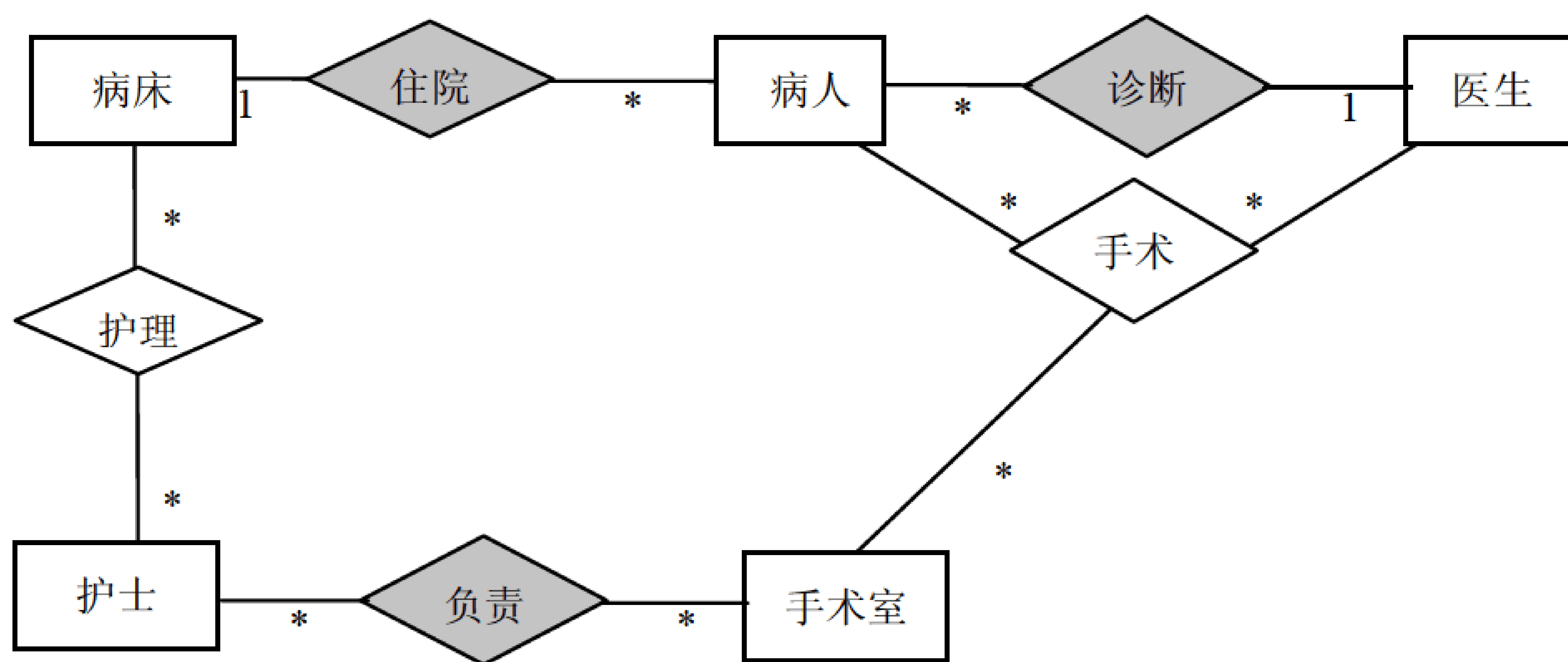
【问题 3】

本问题考查数据库的概念结构设计，根据新增的需求新增实体联系图中的实体及联系和联系的类型。

根据问题描述，系统需记录医生给病人开处方的药品信息，则需新增“药品”实体，并在病人实体与医生实体和药品实体三者之间存在“处方”联系，联系的类型是多对多对多(*:*:*)。

参考答案

【问题 1】



【问题 2】

（1）病区，护士号

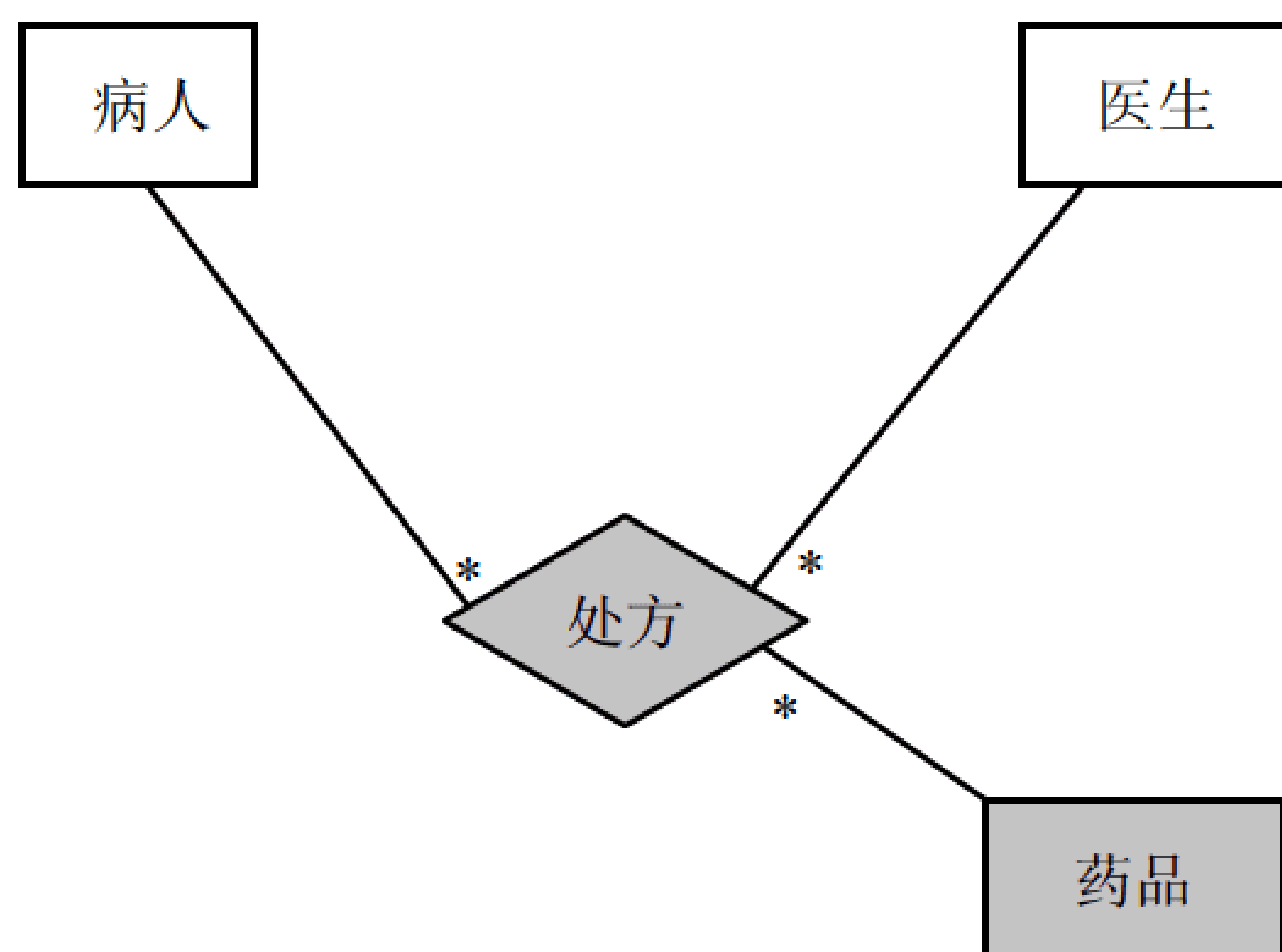
（2）手术室号，护士号，责任

（3）病案号，病床号

（4）病案号，医生编号

（5）病案号，手术室号，手术时间，医生编号

【问题 3】



试题三（共 15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某网上购物平台的主要功能如下：

（1）创建订单。顾客（Customer）在线创建订单（Order），主要操作是向订单中添加项目、从订单中删除项目。订单中应列出所订购的商品（Product）及其数量（quantities）。

（2）提交订单。订单通过网络来提交。在提交订单时，顾客需要提供其姓名（name）、收货地址（address）以及付款方式（form of payment）（预付卡、信用卡或者现金）。为了制定送货计划以及安排送货车辆，系统必须确定订单量（volume）。除此之外，还必须记录每种商品的名称（name）、进价（cost price）、售价（sale price）以及单件商品的包装体积（cubic volume）。

（3）处理订单。订单处理人员接收来自系统的订单；根据订单内容，安排配货，制定送货计划。在送货计划中不仅要指明发货日期（delivery date），还要记录每个订单的限时发送要求（Delivery Time Window）。

（4）派单。订单处理人员将已配好货的订单转交给派送人员。

（5）送货/收货。派送人员将货物送到顾客指定的收货地址。当顾客收货时，需要在运货单（delivery slip）上签收。签收后的运货单最终需交还给订单处理人员。

（6）收货确认。当订单处理人员收到签收过的运货单后，会和顾客进行一次再确认。

现采用面向对象方法开发上述系统，得到如图 3-1 所示的用例图和图 3-2 所示的类图。

【问题 1】（5 分）

根据说明中的描述，给出图 3-1 中 A1～A3 所对应的参与者名称和 U1～U2 处所对

应的用例名称。

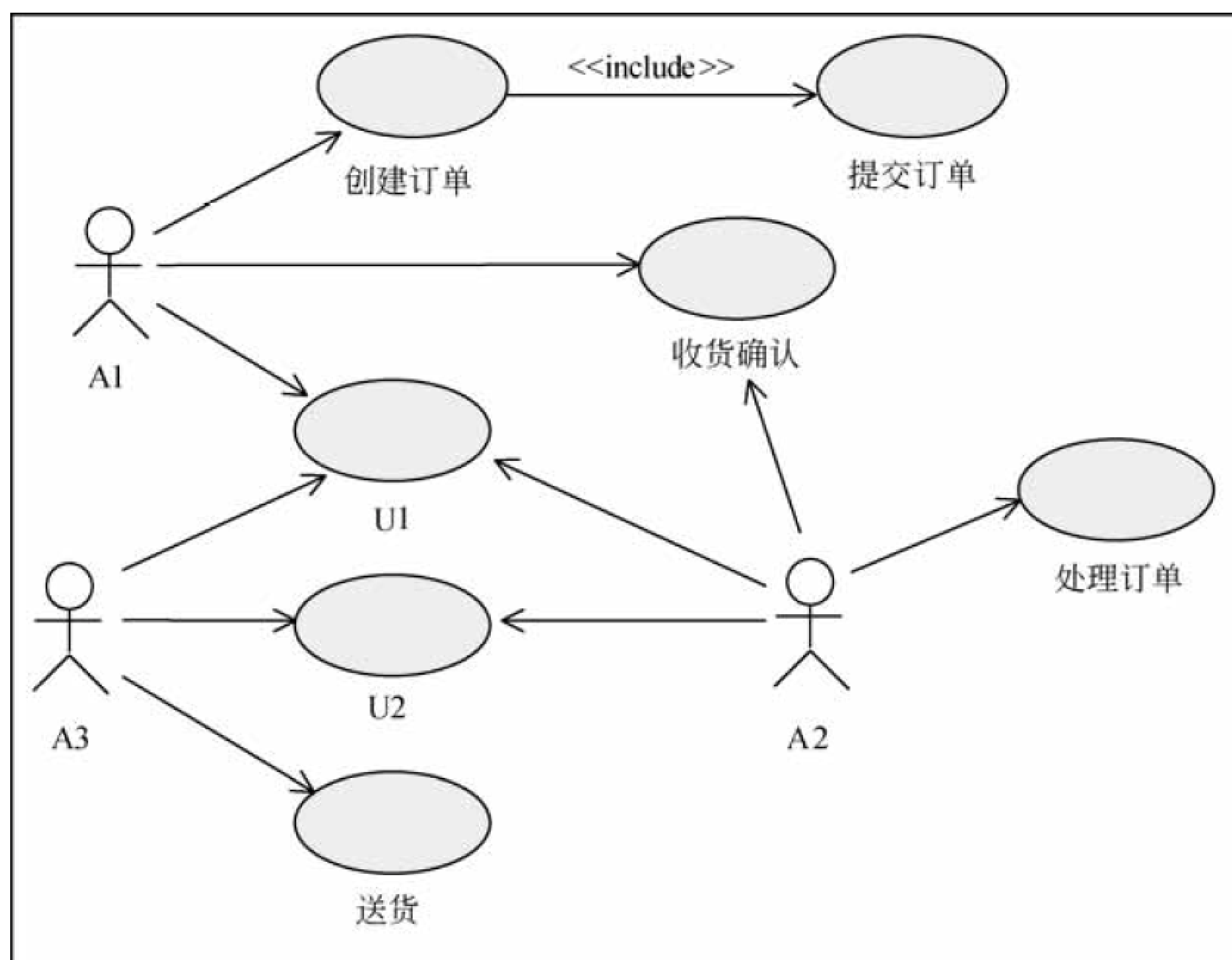


图 3-1　用例图

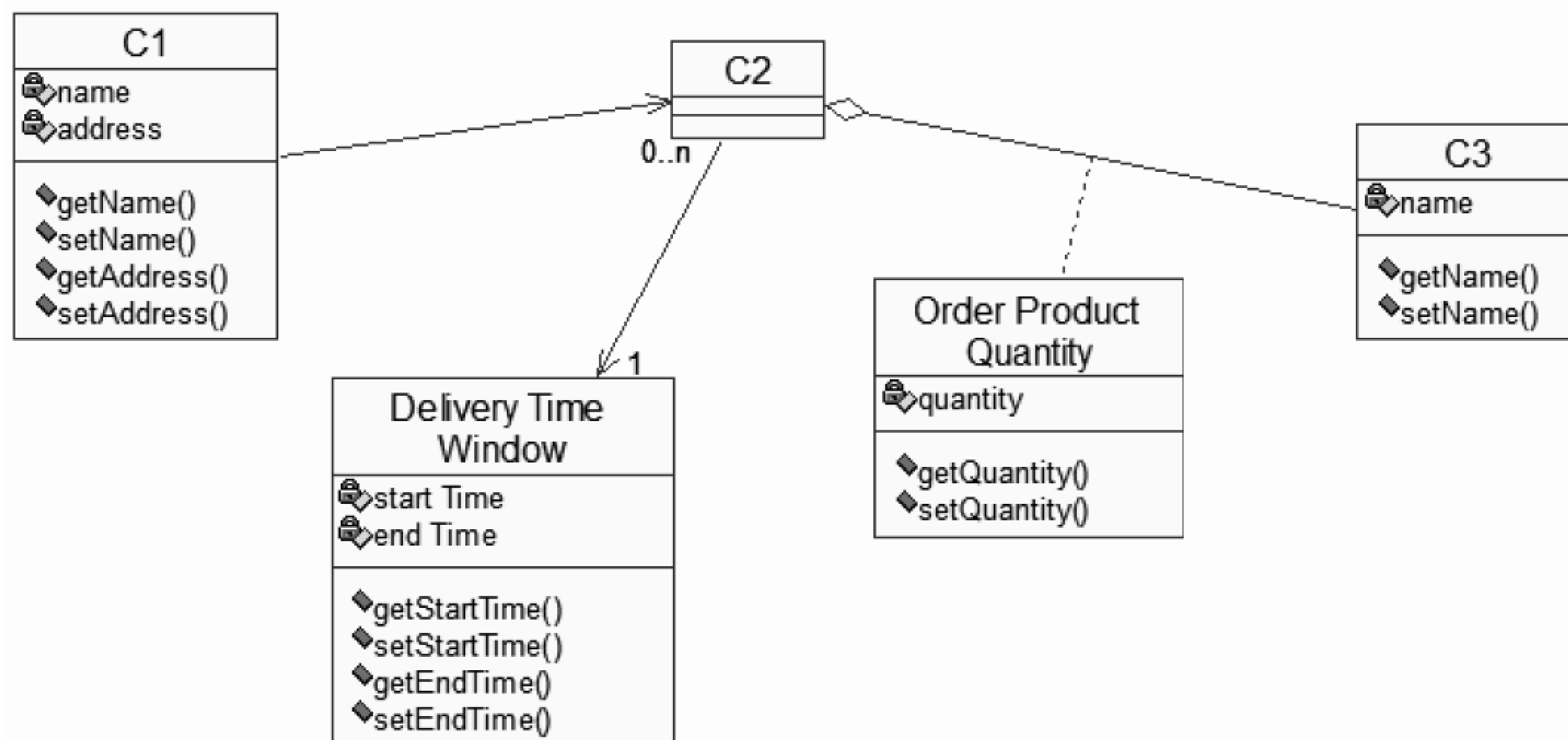


图 3-2　类图

【问题 2】（7 分）

根据说明中的描述，给出图 3-2 中 C1～C3 所对应的类名以及（1）～（4）处所对

应的多重度（类名使用说明中给出的英文词汇）。

【问题 3】（3 分）

根据说明中的描述，将类 C2 和 C3 的属性补充完整（属性名使用说明中给出的英文词汇）。

试题三分析

本题属于经典的考题，主要考查面向对象分析方法以及UML的用例图和类图的相关知识。

【问题 1】

本问题要求将图 3-1 所给出的用例图补充完整。用例图的构成要素有参与者（Actor）、用例（Use Case）以及用例之间的关系。由图 3-1 可知，题目中“网上购物平台”的用例模型中共有 3 个参与者、7 个用例。图中给出了其中的 5 个用例，需要补充所缺少的两个用例和三个参与者。解答此题时，应首先确定参与者，然后再找到与每个参与者对应的用例。

参与者表示要与本系统发生交互的一个角色单元。与系统交互的外部人员、角色、其他的计算机系统、物理实体等通常都可以看作是参与者。从“说明”中可以看出，在本题的描述中出现了“顾客”“订单处理人员”“派送人员”这三种角色。而用例图中恰好缺少了三个参与者，所以可以确定这三种角色就应该与 A1~A3 对应。接下来要确定的就是它们之间的对应关系，这就需要明确参与者与用例之间的关系了。解答时可以依据用例图逐个考查其中的参与者。

先从 A1 开始。A1 参与的用例分别为“创建订单”、“收货确认”以及需要补充的用例 U1。用例总是由参与者启动的。以“创建订单”“收货确认”为关键词，在“说明”中查找出现这两个关键词的语句的主语。这个主语就是该用例的参与者。由“说明”可见，A1 应该表示的是“顾客”。采用同样的方法，可以确定 A2 表示“订单处理人员”，A3 表示“派送人员”。

由于缺少的用例 U1 和 U2 与参与者 A1 和 A3 相关，因此重点考查“说明”中出现了“顾客”和“派送人员”的语句。比对“说明”和用例图可知，功能“派单”“收货”没有出现在用例图中。“订单处理人员”和“派送人员”都参与了“派单”功能，所以 U2 处应该是用例“派单”。而“收货”功能涉及了三种角色：派送人员、顾客（两者完成货物的交接）、订单处理人员（收回顾客签字后的运货单，此时一次完整的收货活动才算结束），所以 U1 处应该是“收货”。

【问题 2】

本问题考查的是类图建模。解题的重点在于根据类图中提供的类、类的属性以及类之间的关联关系推断出需要补充的类。

先看类 C1，C1 的关键属性都已经给出了。由“说明”可知，属性 address 表示的是“收货地址”。而收货地址的最初始来源是顾客，所以类 C1 代表的应该是 Customer

（顾客）。

另外，从“说明”中可知，“订单”和“商品”是“网上购物平台”系统的关键概念，所以应该在类图上有相应的类来表示。由此可以推断出，C2 和 C3 就应该与这两个概念相关。观察类图，在 C2 和 C3 之间存在着一个关联类 Order Product，而 C3 和 C2 之间又存在着一个聚集关系，C3 是构成 C2 的部分对象。根据“说明”，订单中包含了订购的商品，所以 C2 应该是 Order（订单），C3 应该是 Product（商品）。

三个类都确定之后，下面来识别它们之间的多重度。

首先顾客（C1）和订单（C2）之间的关联，一名顾客可以在线创建多个订单，但是一个订单只能由一名顾客来创建。所以（1）处应填写“1”，（2）处应填写“0..n”。

订单（C2）和商品（C3），一个订单上可以订购多件商品，而一件商品可以出现在一个或多个订单中，也可能没有任何顾客订购某种商品。因此 C2 和 C3 之间是一种多对多联系。在图 3-2 中，采用了关联类的方法来表示多对多联系。这里（3）处应填写“0..n”，（4）处应填写“1..n”。这里下限 1 表示一个订单中至少应包含一件商品。

【问题 3】

本问题考查类的关键属性的识别。由“说明”中给出的描述可知，类C2的属性至少应包括 volume、delivery date、form of payment；类 C3 的属性至少应包括 cubic volume、cost price、sale price。

参考答案

【问题 1】

A1：顾客　　A2：订单处理人员　　A3：派送人员

U1：收货　　U2：派单

【问题 2】

C1：Customer　　C2：Order　　C3：Product

（1）1　　（2）0..n 或 0..*　　（3）0..n 或 0..*　　（4）1..n 或 1..*

【问题 3】

C2：volume、delivery date、form of payment

C3：cubic volume、cost price、sale price

试题四（共 15 分）

阅读下列说明和 C 代码，回答问题 1 至问题 3，将解答写在答题纸的对应栏内。

【说明】

用两台处理机 A 和 B 处理 n 个作业。设 A 和 B 处理第 i 个作业的时间分别为 a_i 和 b_i。由于各个作业的特点和机器性能的关系，对某些作业，在 A 上处理时间长，而对某些作业在 B 上处理时间长。一台处理机在某个时刻只能处理一个作业，而且作业处理是不可中断的，每个作业只能被处理一次。现要找出一个最优调度方案，使得 n 个作业被这两台处理机处理完毕的时间（所有作业被处理的时间之和）最少。

算法步骤：

（1）确定候选解上界为最短的单台处理机处理所有作业的完成时间 m，$m = \min\left(\sum_{i=1}^{n} a_i, \sum_{i=1}^{n} b_i\right)$。

（2）用 p(x,y,k)=1 表示前 k 个作业可以在 A 用时不超过 x 且在 B 用时不超过 y 时间内处理完成，则 $p(x,y,k)=p(x-a_k,y,k-1) \| p(x,y-b_k,k-1)$ （||表示逻辑或操作）。

（3）得到最短处理时间为 min(max(x,y)) 。

【C 代码】

下面是该算法的 C 语言实现。

（1）常量和变量说明

n：作业数

m：候选解上界

a：数组，长度为 n，记录 n 个作业在 A 上的处理时间，下标从 0 开始

b：数组，长度为 n，记录 n 个作业在 B 上的处理时间，下标从 0 开始

k：循环变量

p：三维数组，长度为(m+1)* (m+1)* (n+1)

temp：临时变量

max：最短处理时间

（2）C 代码

```
#include <stdio.h>
int n, m;
int a[60], b[60], p[100][100][60];
void read(){  /*输入 n、a、b，求出 m，代码略*/  }
void schedule(){          /*求解过程*/
    int x,y,k;
    for(x = 0; x <= m; x++){
        for (y = 0; y <= m; y++){
            ____(1)____;
            for(k = 1; k <= n; k++)
                p[x][y][k] = 0;
        }
    }
    for(k = 1;k <= n;k++){
        for (x = 0;x <= m;x++){
            for(y = 0; y <= m; y++){
                if(x - a[k - 1] >= 0)  ____(2)____;
                if(____(3)____)  p[x][y][k] = (p[x][y][k] || p[x][y - b[k -
```

```
                    1]][k - 1]);
                }
            }
        }
    }
    void write(){          /*确定最优解并输出*/
        int x,y, temp,max = m;
        for(x = 0;x <= m;x++){
            for (y = 0;y <= m;y++){
                if( ____(4)____ ){
                    temp = ____(5)____ ;
                    if(temp < max)     max = temp;
                }
            }
        }
        printf("\n%d\n", max);
    }
    void main(){    read();    schedule();    write();    }
```

【问题 1】（9 分）

根据以上说明和 C 代码，填充 C 代码中的空（1）～（5）。

【问题 2】（2 分）

根据以上 C 代码，算法的时间复杂度为 （6） （用 O 符号表示）。

【问题 3】（4 分）

考虑 6 个作业的实例，各个作业在两台处理机上的处理时间如表 4-1 所示。该实例的最优解为 （7） ，最优解的值（即最短处理时间）为 （8） 。最优解用$(x_1,x_2,x_3,x_4,x_5,x_6)$表示，其中若第 i 个作业在 A 上处理，则 $x_i=1$，否则 $x_i=2$。如（1,1,1,1,2,2）表示作业 1,2,3 和 4 在 A 上处理，作业 5 和 6 在 B 上处理。

表 4-1

	作业 1	作业 2	作业 3	作业 4	作业 5	作业 6
处理机 A	2	5	7	10	5	2
处理机 B	3	8	4	11	3	4

试题四分析

本题考查算法设计与分析技术以及算法的 C 语言实现，是比较传统的题目，要求考生细心分析题目中所描述的内容。

【问题 1】

schedule 函数计算 p 数组中元素的值。第一个三重 for 循环进行 p 数组的初始化，当

k = 0 时，p[i][j][k] = 1；而当 k≠0 时，p[i][j][k] = 0，因此(1)处应填“p[x][y][0] = 1”。

在接下来的三重 for 循环中，自底向上计算 p[i][j][k]的值，计算根据题中定义来进行，(2)处应填“p[x][y][k] = p[x – a[k – 1]][y][k – 1]”。对应于上一行的条件，(3)处应填“y – b [k – 1] >= 0”。

write 函数中，已经计算出 p 数组元素的值，因此此时每一个 p[i][j][n] = 1 表示一个可行解，因此(4)处应填“p[x][y][n] == 1”或等价的表达式。其中该解对应的处理时间为 i 和 j 中的较大值，因此(5)处应填“(x >= y)? x:y”。在所有的可行解中，确定最小的处理时间即为问题的最优解。

【问题 2】

根据上述 C 代码，函数 schedule 有两处三重 for 循环，时间复杂度为 $O(m^2n)$。函数 write 有一处两重 for 循环，时间复杂度为 $O(m^2)$。因此整个算法的时间复杂度为 $O(m^2n + m^2)$，由于 m^2 在数量级上小于 m^2n，因此算法的时间复杂度可以表示为 $O(m^2n)$。

【问题 3】

根据题意和算法，可以得到若作业 1、2、5 和 6 在处理机 A，而作业 3 和 4 在处理机 B 上处理，可以得到最优解。此时在处理机 A 上的处理时间为14，而在处理机 B 上的处理时间为 15，因此最优解的值，即最短的处理时间为 15，而最优解为（1,1,2,2,1,1）。

参考答案

【问题 1】

（1）p[x][y][0] = 1

（2）p[x][y][k] = p[x – a[k – 1]][y][k – 1]

（3）y – b[k – 1] >= 0

（4）p[x][y][n] == 1 或 p[x][y][n] 或 p[x][y][n] != 0

（5）(x >= y)? x:y

【问题 2】

（6）$O(m^2n)$

【问题 3】

（7）(1,1,2,2,1,1)

（8）15

试题五（共 15 分）

阅读下列说明和 C++代码，将应填入 __（n）__ 处的字句写在答题纸的对应栏内。

【说明】

某咖啡店售卖咖啡时，可以根据顾客的要求在其中加入各种配料，咖啡店会根据所加入的配料来计算费用。咖啡店所供应的咖啡及配料的种类和价格如下表所示。

咖啡	价格/杯（￥）	配料	价格/份（￥）
蒸馏咖啡（Espresso）	25	摩卡（Mocha）	10
深度烘焙咖啡（DarkRoast）	20	奶泡（Whip）	8

现采用装饰器（Decorator）模式来实现计算费用的功能，得到如图 5-1 所示的类图。

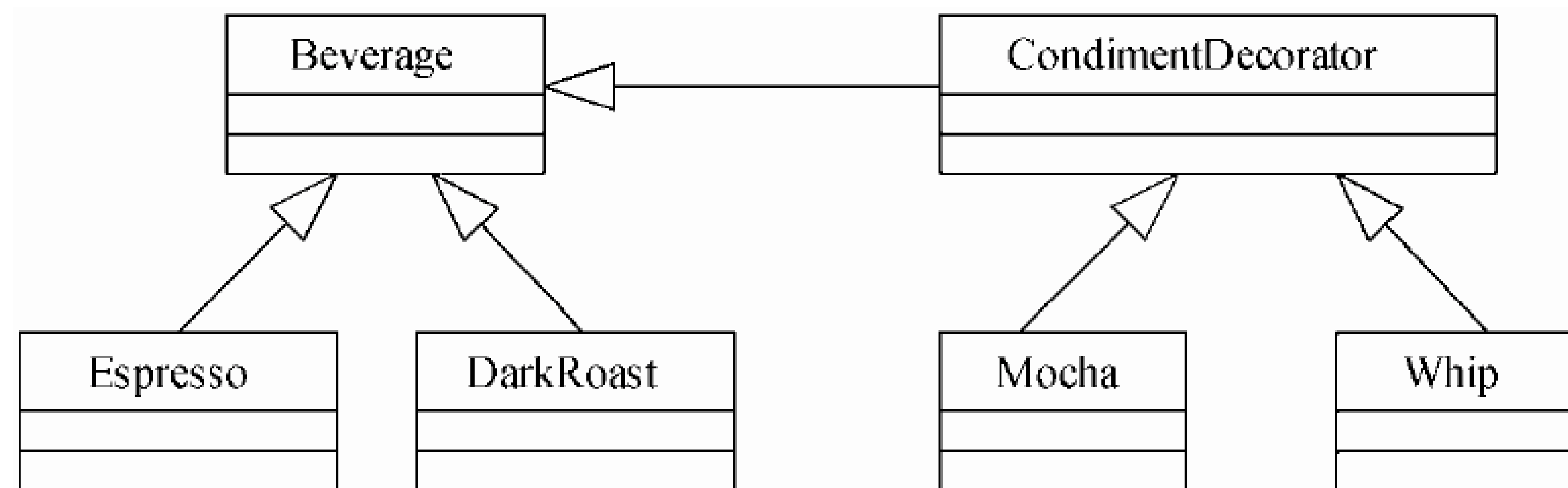


图 5-1　类图

【C++代码】

```
#include <iostream>
#include <string>
using namespace std;
const int ESPRESSO_PRICE = 25;
const int DRAKROAST_PRICE = 20;
const int MOCHA_PRICE = 10;
const int WHIP_PRICE = 8;
class Beverage {    // 饮料
__(1)__:    string description;
public:
__(2)__ () {  return description;  }
    __(3)__ ;
};
class CondimentDecorator : public Beverage {   // 配料
protected:
    __(4)__ ;
};
class Espresso : public Beverage {    // 蒸馏咖啡
public:
    Espresso() { description = "Espresso";  }
    int cost() { return ESPRESSO_PRICE; }
};
class DarkRoast : public Beverage {   // 深度烘焙咖啡
public:
```

```
        DarkRoast() { description = "DardRoast"; }
        int cost() { return DRAKROAST_PRICE; }
    };
    class Mocha : public CondimentDecorator {  // 摩卡
    public:
        Mocha(Beverage* beverage) {  this->beverage = beverage; }
        string getDescription()  {  return beverage->getDescription()+ ",
Mocha";  }
        int cost() { return MOCHA_PRICE + beverage->cost(); }
    };
    class Whip : public CondimentDecorator {  // 奶泡
    public:
        Whip(Beverage* beverage) {  this->beverage = beverage; }
        string getDescription()  { return beverage->getDescription()+ ",
Whip";  }
        int cost() { return WHIP_PRICE + beverage->cost(); }
    };

    int main() {
        Beverage* beverage = new DarkRoast();
        beverage = new Mocha(__(5)__);
        beverage= new Whip(__(6)__);
        cout << beverage->getDescription() << " ￥" << beverage->cost() << endl;
        return 0;
    }
```

编译运行上述程序，其输出结果为：

```
DarkRoast, Mocha, Whip ￥38
```

试题五分析

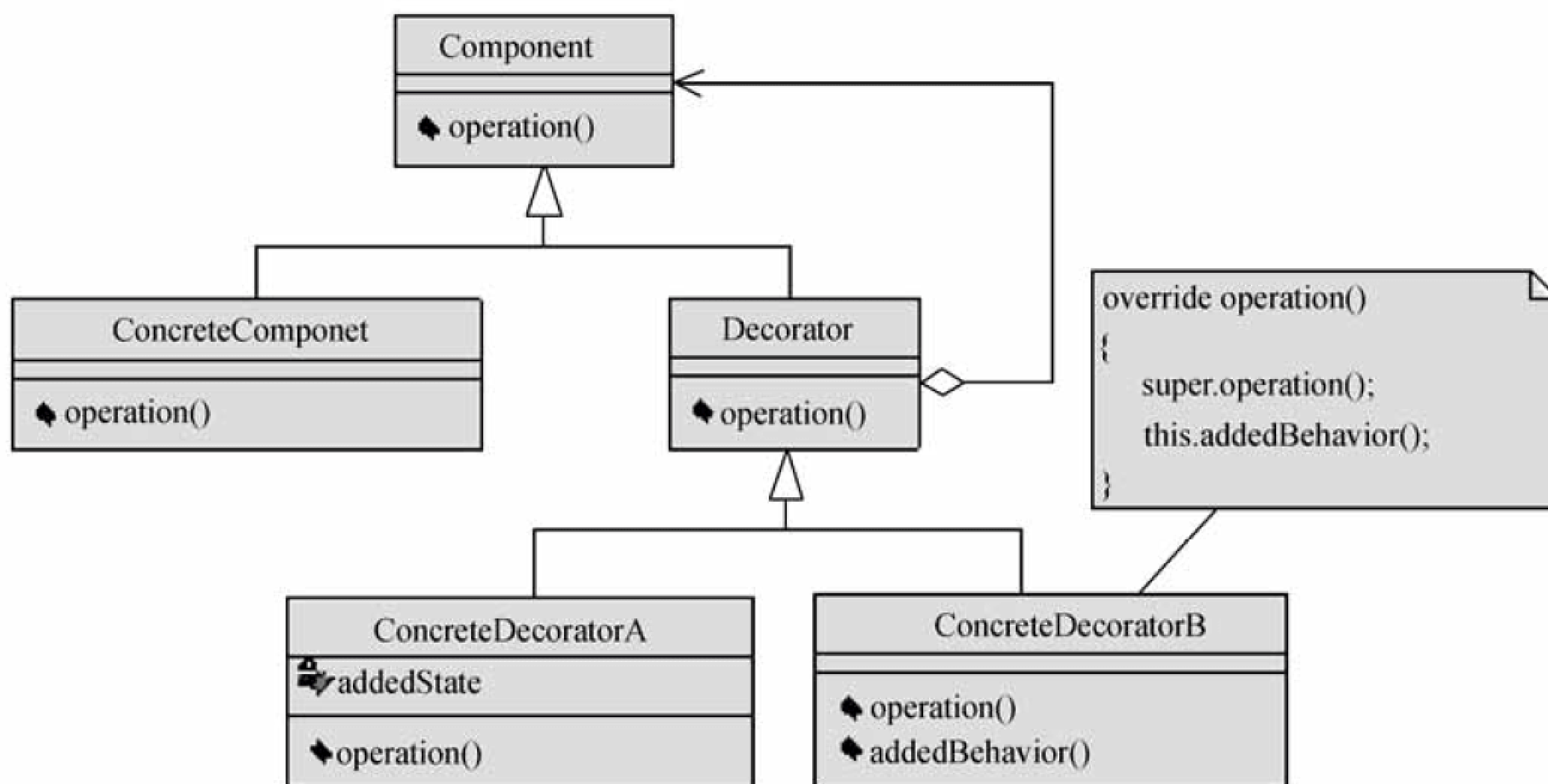
本题考查装饰器（Decorator）模式的概念及应用。

Decorator 模式动态地给一个对象添加一些额外的职责，就增加功能来说，装饰模式提供了比继承更有弹性的替代方案。

Decorator 模式的优点是有效避免了使用继承方式扩展对象功能而带来的灵活性差、子类无限制扩展的问题；装饰者与被装饰者之间虽然都是同一类型，但是它们彼此是完全独立并可以独立任意改变的。

Decorator 模式的适用场合是：想透明并且动态地给对象增加新的职责；给对象增加的职责，在未来存在增加或减少的可能。

Decorator 模式的类图如下所示：



题目利用Decorator 模式来计算各种配料组合的咖啡的价格。Beverage 相当于抽象的 Component 类，最终要计算出 Beverage 的价钱。Espresso 和 DarkRoast 是 4 个具体的组件，代表一种咖啡类型。Macha 和 Whip 是配料装饰者，可以添加到不同类型的咖啡中。CondimentDecorator 相当于 Decorator，是装饰者共同实现的接口。在本题中，确定装饰者共同实现的接口是什么，是一个重要的考查点。

下面来分析程序。

第（1）空要求给出类 Beverage 的数据成员 description 的访问控制。C++语言提供了 3 种访问控制 private、public 和 protected。private 成员只能在定义它的类中或其友元访问；public 成员可以被任意访问；protected 成员只能在定义它的类及其子类中访问。从程序上下文可知，在 Beverage 的子类 Espresso 和 DarkRoast 中都需要访问 description，所以第（1）空应填“protected”。

第（2）、（3）空要求确定 Beverage 中提供的公共接口。解答时应全面阅读程序，通过子类的代码来推断父类所提供的接口是什么。

首先来观察 Beverage 的两个子类。在 Espresso 和 DarkRoat 中都出现的成员函数是 cost，其功能是给出咖啡的价格；而在这两个类中，cost 的实现代码又不相同。这就提示我们这里采用了 C++语言的动态多态机制，意味着需要在这两个类的父类中定义一个虚拟函数，其函数原型就应该是 int cost()。现在回到第（2）、（3）空。第（2）空给出了成员函数的实现体，说明（2）处的成员函数是与 description 相关的。也就是说，第（2）空处的成员函数不可能是 cost。这样就可以确定第（3）空应该是 cost 成员函数。仔细观察第（3）空，这条语句的结束符是“;”，即第（3）空中只能出现函数的接口声明。只有接口声明的虚函数，在 C++中只能是纯虚拟函数。所以第（3）空应填写“virtual int cost() = 0”。

如何来确定第（2）空？在类Espresso和DarkRoast中已经找不到相关信息了，我们考查剩余的类。

类CondimentDecorator是Beverage的子类，那么第（2）和（3）空处的成员函数都会被它继承。而类Mocha和Whip又是CondimentDecorator的子类，第（2）和（3）空处的成员函数同样也会被这两个类继承。观察Mocha和Whip的代码，可以发现在这两个子类中也重置了纯虚拟函数cost。除此之外还可以发现，在这两个子类中都出现了成员函数string getDescription()，并且其在两个类中的实现代码也不相同。这说明，getDescription也应该是父类中所定义的虚拟函数。从子类沿着继承路径追溯到父类，可以确定第（2）空就应该是虚拟函数getDescription最初的定义之处。所以第（2）空应填写“virtual string getDescription”。

第（4）空考查的是类CondimentDecorator中定义的数据成员，要求确定其类型和名称。由于声明成了protected，因此在其子类Mocha和Whip中可以找到答案。在这两个类中都出现了“this–>beverage”表达式，说明beverage应该是该类的数据成员。但在这两个类中都没有定义数据成员，很显然，这个数据成员应该是从父类继承而来的。所以第（4）空应填“Beverage* beverage”。

第（5）和（6）空考查的是Decorator模式的使用方法。从主函数可知，其基本过程是现制作一杯烘焙咖啡（DarkRoast），然后再向这杯咖啡中加入1份Mocha和1份Whip，最后这杯咖啡的价格应该是烘焙咖啡的价格+1份Mocha的价格+1份Whip的价格。（5）和（6）处都应该填写“beverage”。

参考答案

（1）protected

（2）virtual string getDescription

（3）virtual int cost() = 0

（4）Beverage* beverage

（5）beverage

（6）beverage

试题六（共15分）

阅读下列说明和Java代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

某咖啡店售卖咖啡时，可以根据顾客的要求在其中加入各种配料，咖啡店会根据所加入的配料来计算费用。咖啡店所供应的咖啡及配料的种类和价格如下表所示。

咖啡	价格/杯（￥）	配料	价格/份（￥）
蒸馏咖啡（Espresso）	25	摩卡（Mocha）	10
深度烘焙咖啡（DarkRoast）	20	奶泡（Whip）	8

现采用装饰器（Decorator）模式来实现计算费用的功能，得到如图 6-1 所示的类图。

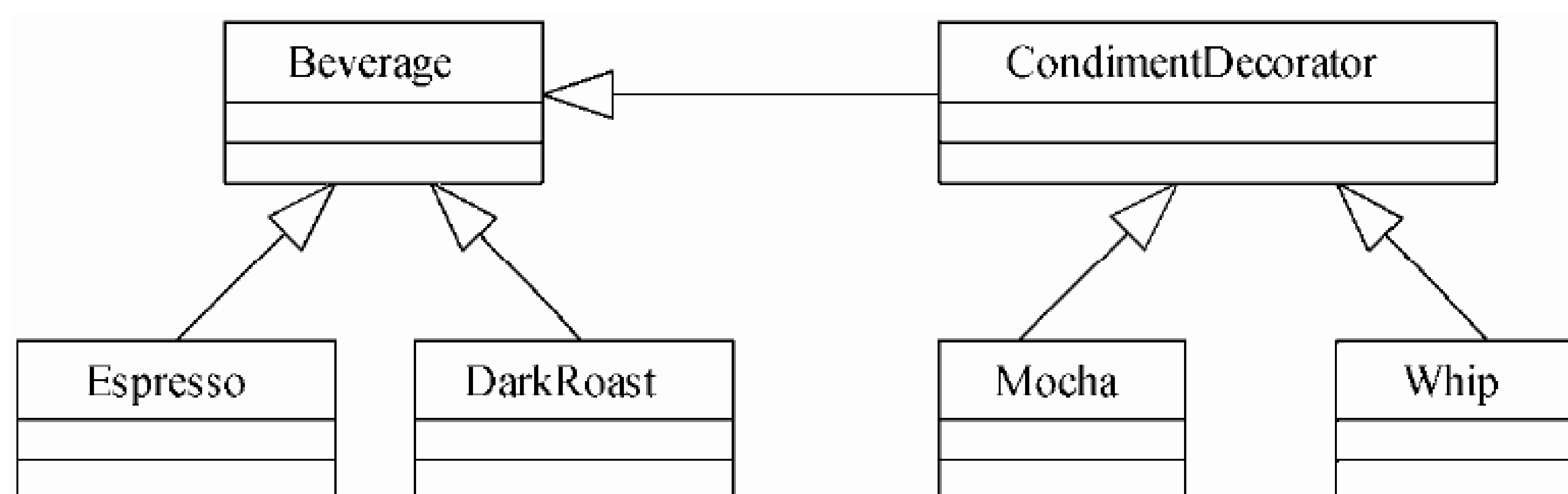


图 6-1　类图

【Java 代码】

```
import java.util.*;
___(1)___ class Beverage {      // 饮料
  String description = "Unknown Beverage";
  public ___(2)___ () {      return description;  }
  public ___(3)___ ;
}
abstract class CondimentDecorator extends Beverage {   // 配料
  ___(4)___ ;
}
class Espresso extends Beverage {     // 蒸馏咖啡
  private final int ESPRESSO_PRICE = 25;
  public Espresso() {   description = "Espresso";  }
  public int cost() {    return ESPRESSO_PRICE;  }
}

class DarkRoast extends Beverage {    // 深度烘焙咖啡
  private final int DARKROAST_PRICE = 20;
  public DarkRoast() {   description = "DarkRoast";  }
  public int cost() {    return DARKROAST_PRICE;      }
}
class Mocha extends CondimentDecorator {   // 摩卡
  private final int MOCHA_PRICE = 10;
  public Mocha(Beverage beverage) {
     this.beverage = beverage;
  }
  public String getDescription() {
     return beverage.getDescription() + ", Mocha";
  }
```

```
    public int cost() {
        return MOCHA_PRICE + beverage.cost();
    }
  }
  class Whip extends CondimentDecorator {  // 奶泡
    private final int WHIP_PRICE = 8;
    public Whip(Beverage beverage) {  this.beverage = beverage; }
    public String getDescription()  {
          return beverage.getDescription()+ ", Whip";
    }
    public int cost() { return WHIP_PRICE + beverage.cost(); }
  }

  public class Coffee {
    public static void main(String args[]) {
      Beverage beverage = new DarkRoast();
      beverage = new Mocha( （5） );
      beverage = new Whip( （6） );
      System.out.println(beverage.getDescription() + " ￥" +
      beverage.cost());
    }
  }
```

编译运行上述程序，其输出结果为：

```
DarkRoast, Mocha, Whip ￥38
```

试题六分析

本题考查装饰器（Decorator）模式的概念及应用。

Decorator 模式动态地给一个对象添加一些额外的职责，就增加功能来说，装饰模式提供了比继承更有弹性的替代方案。

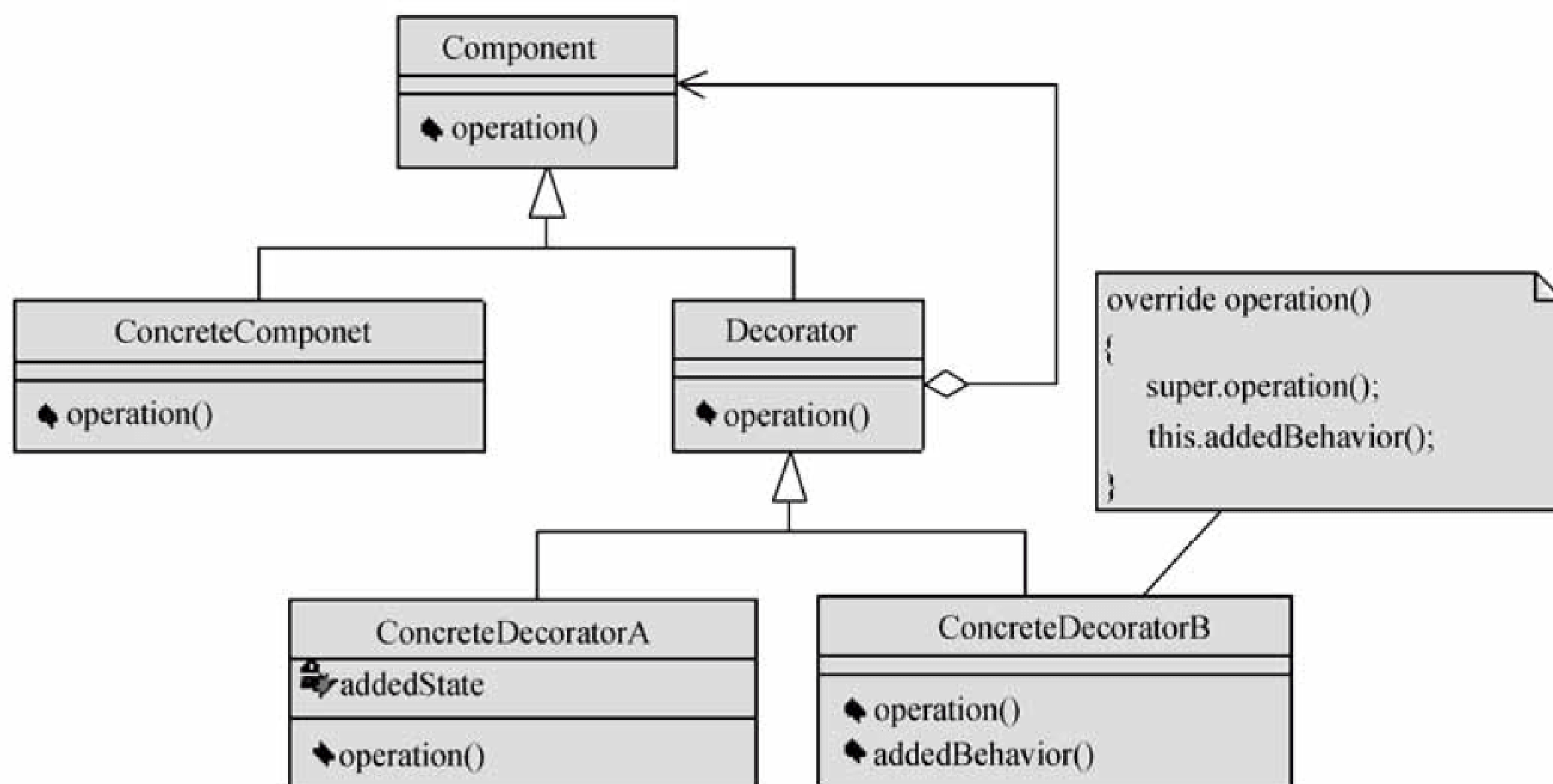
Decorator 模式的优点是有效避免了使用继承方式扩展对象功能而带来的灵活性差、子类无限制扩展的问题；装饰者与被装饰者之间虽然都是同一类型，但是它们彼此是完全独立并可以独立任意改变的。

Decorator 模式的适用场合是：想透明并且动态地给对象增加新的职责；给对象增加的职责，在未来存在增加或减少的可能。

Decorator 模式的类图如下所示：

题目利用Decorator 模式来计算各种配料组合的咖啡的价格。Beverage 相当于抽象的 Component 类，最终要计算出 Beverage 的价钱。Espresso 和 DarkRoast 是 4 个具体的组件，代表一种咖啡类型。Macha 和 Whip 是配料装饰者，可以添加到不同类型的咖啡中。

CondimentDecorator 相当于 Decorator，是装饰者共同实现的接口。在本题中，确定装饰者共同实现的接口是什么，是一个重要的考查点。



下面来分析程序。

Decorator 模式中的 Component 通常都用抽象类来实现。所以第（1）空应填写“abstract”。

第（2）、（3）空要求确定 Beverage 中提供的公共接口。解答时应全面阅读程序，通过子类的代码来推断父类所提供的接口是什么。

首先来观察 Beverage 的两个子类。在 Espresso 和 DarkRoat 中都出现的成员函数是 cost，其功能是给出咖啡的价格；而在这两个类中，cost 的实现代码又不相同。这意味着需要在这两个类的父类中定义一个抽象函数，其函数原型就应该是 int cost()。现在回到第(2)、(3)空。第(2)空给出了成员函数的实现体，说明(2)处的成员函数是与 description 相关的。也就是说，第（2）空处的成员函数不可能是 cost。这样就可以确定第（3）空应该是 cost 成员函数。所以第（3）空应填写“abstract int cost()”。

如何来确定第（2）空？在类 Espresso 和 DarkRoast 中已经找不到相关信息了，我们考查剩余的类。

类 CondimentDecorator 是 Beverage 的子类，那么第（2）和（3）空处的成员函数都会被它继承。而类 Mocha 和 Whip 又是 CondimentDecorator 的子类，第（2）和（3）空处的成员函数同样也会被这两个类继承。观察 Mocha 和 Whip 的代码，可以发现在这两个子类中也重置了抽象函数 cost。除此之外还可以发现，在这两个子类中都出现了成员函数 String getDescription()。从子类沿着继承路径追溯到父类，可以确定第（2）空就应该是成员函数 getDescription 最初的定义之处。所以第（2）空应填写“String getDescription”。

第（4）空考查的是类 CondimentDecorator 中定义的数据成员，要求确定其类型和名称。由于声明成了 protected，因此在其子类 Mocha 和 Whip 中可以找到答案。在这两个类中都出现了“this.beverage”表达式，说明 beverage 应该是该类的数据成员。但在这两个类中都没有定义数据成员，很显然，这个数据成员应该是从父类继承而来的。所以第（4）空应填“Beverage beverage”。

第（5）和（6）空考查的是Decorator 模式的使用方法。从主函数可知，其基本过程是现制作一杯烘焙咖啡（DarkRoast），然后再向这杯咖啡中加入 1 份 Mocha 和 1 份 Whip，最后这杯咖啡的价格应该是烘焙咖啡的价格+1 份 Mocha 的价格+1 份 Whip 的价格。（5）和（6）处都应该填写“beverage”。

参考答案

（1）abstract

（2）String getDescription

（3）abstract int cost()

（4）Beverage beverage

（5）beverage

（6）beverage

第 15 章　2012 下半年软件设计师上午试题分析与解答

试题（1）

在 CPU 中，__(1)__不仅要保证指令的正确执行，还要能够处理异常事件。

（1）A．运算器　　B．控制器　　C．寄存器组　　D．内部总线

试题（1）分析

本题考查计算机系统硬件方面的基础知识。

计算机中的 CPU 是硬件系统的核心，用于数据的加工处理，能完成各种算术、逻辑运算及控制功能。其中，控制器的作用是控制整个计算机的各个部件有条不紊地工作，它的基本功能就是从内存取指令和执行指令。

参考答案

（1）B

试题（2）

循环冗余校验码（CRC）利用生成多项式进行编码。设数据位为 k 位，校验位为 r 位，则 CRC 码的格式为__(2)__。

（2）A．k 个数据位之后跟 r 个校验位

B．r 个校验位之后跟 k 个数据位

C．r 个校验位随机加入 k 个数据位中

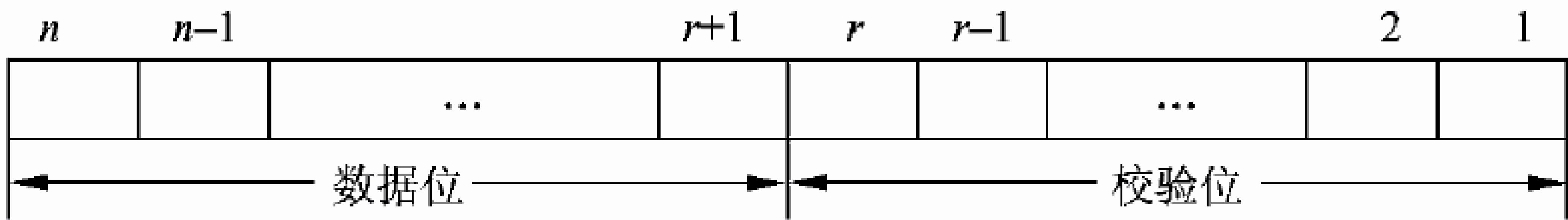
D．r 个校验位等间隔地加入 k 个数据位中

试题（2）分析

本题考查数据校验基础知识。

计算机系统运行时，各个部件之间要进行数据交换，为了确保数据在传送过程中正确无误，一是提高硬件电路的可靠性；二是提高代码的校验能力，包括查错和纠错。常用的三种校验码：奇偶校验码（Parity Codes）、海明码（Hamming Code）和循环冗余校验（Cyclic Redundancy Check，CRC）码。

循环冗余校验码广泛应用于数据通信领域和磁介质存储系统中。它利用生成多项式为 k 个数据位产生 r 个校验位来进行编码，其编码长度为 $k+r$。CRC 的代码格式为：



参考答案

（2）A

试题（3）

以下关于数的定点表示和浮点表示的叙述中，不正确的是__（3）__。

（3）A．定点表示法表示的数（称为定点数）常分为定点整数和定点小数两种

B．定点表示法中，小数点需要占用一个存储位

C．浮点表示法用阶码和尾数来表示数，称为浮点数

D．在总位数相同的情况下，浮点表示法可以表示更大的数

试题（3）分析

本题考查数据表示基础知识。

各种数据在计算机中表示的形式称为机器数，其特点是采用二进制计数制，数的符号用 0、1 表示，小数点则隐含表示而不占位置。机器数对应的实际数值称为数的真值。

为了便于运算，带符号的机器数可采用原码、反码、补码和移码等不同的编码方法。

所谓定点数，就是表示数据时小数点的位置固定不变。小数点的位置通常有两种约定方式：定点整数（纯整数，小数点在最低有效数值位之后）和定点小数（纯小数，小数点在最高有效数值位之前）。

当机器字长为 n 时，定点数的补码和移码可表示 2^n 个数，而其原码和反码只能表示 2^n－1 个数（0 表示占用了两个编码），因此，定点数所能表示的数值范围比较小，运算中很容易因结果超出范围而溢出。

数的浮点表示的一般形式为：$N=2^E \times F$，其中 E 称为阶码，F 为尾数。阶码通常为带符号的纯整数，尾数为带符号的纯小数。浮点数的表示格式如下：

阶符	阶码	数符	尾数

很明显，一个数的浮点表示不是唯一的。当小数点的位置改变时，阶码也相应改变，因此可以用多种浮点形式表示同一个数。

浮点数所能表示的数值范围主要由阶码决定，所表示数值的精度则由尾数决定。

参考答案

（3）B

试题（4）

__（4）__不属于按寻址方式划分的一类存储器。

（4）A．随机存储器 B．顺序存储器 C．相联存储器 D．直接存储器

试题（4）分析

本题考查存储系统的基础知识。

存储系统中的存储器，按访问方式可分为按地址访问的存储器和按内容访问的存储器；按寻址方式分类可分为随机存储器、顺序存储器和直接存储器。

随机存储器（Random Access Memory，RAM）指可对任何存储单元存入或读取数据，访问任何一个存储单元所需的时间是相同的。

顺序存储器（Sequentially Addressed Memory，SAM）指访问数据所需要的时间与数据所在的存储位置相关，磁带是典型的顺序存储器。

直接存储器（Direct Addressed Memory，DAM）是介于随机存取和顺序存取之间的一种寻址方式。磁盘是一种直接存取存储器，它对磁道的寻址是随机的，而在一个磁道内，则是顺序寻址。

相联存储器是一种按内容访问的存储器。其工作原理就是把数据或数据的某一部分作为关键字，将该关键字与存储器中的每一单元进行比较，从而找出存储器中所有与关键字相同的数据字。

参考答案

（4）C

试题（5）

在 I/O 设备与主机间进行数据传输时，CPU 只需在开始和结束时作少量处理，而无须干预数据传送过程的是__（5）__方式。

（5）A．中断　　B．程序查询　　C．无条件传送　　D．直接存储器存取

试题（5）分析

本题考查计算机系统硬件方面的基础知识。

中断方式下的数据传送是当 I/O 接口准备好接收数据或准备好向 CPU 传送数据时，就发出中断信号通知 CPU。对中断信号进行确认后，CPU 保存正在执行的程序的现场，转而执行提前设置好的 I/O 中断服务程序，完成一次数据传送的处理。这样，CPU 就不需要主动查询外设的状态，在等待数据期间可以执行其他程序，从而提高了 CPU 的利用率。采用中断方式管理 I/O 设备，CPU 和外设可以并行地工作。

程序查询方式下，CPU 通过执行程序查询外设的状态，判断外设是否准备好接收数据或准备好了向 CPU 输入的数据。

直接内存存取（Direct Memory Access，DMA）方式的基本思想是通过硬件控制实现主存与 I/O 设备间的直接数据传送，数据的传送过程由 DMA 控制器（DMAC）进行控制，不需要 CPU 的干预。在 DMA 方式下，由 CPU 启动传送过程，即向设备发出“传送一块数据”的命令，在传送过程结束时，DMAC 通过中断方式通知 CPU 进行一些后续处理工作。

参考答案

（5）D

试题（6）

__（6）__不属于系统总线。

（6）A．ISA　　B．EISA　　C．SCSI　　D．PCI

试题（6）分析

本题考查计算机系统的基础知识。

系统总线又称内总线或板级总线，在微机系统中用来连接各功能部件而构成一个完整的微机系统。系统总线包含有三种不同功能的总线，即数据总线DB（Data Bus）、地址总线AB（Address Bus）和控制总线CB（Control Bus）。

ISA（Industrial Standard Architecture）总线标准是 IBM 公司 1984 年为推出 PC/AT 机而建立的系统总线标准，所以也叫 AT 总线。它是对 XT总线的扩展，以适应 8/16 位数据总线要求。

EISA总线是 1988 年由 Compaq 等 9 家公司联合推出的总线标准。它在 ISA总线的基础上使用双层插座，在原来 ISA 总线的 98 条信号线上又增加了 98 条信号线，也就是在两条 ISA 信号线之间添加一条 EISA 信号线。在实用中，EISA总线完全兼容 ISA 总线信号。

PCI（Peripheral Component Interconnect）总线是当前最流行的总线之一，它是由 Intel 公司推出的一种局部总线。它定义了 32 位数据总线，且可扩展为 64 位。PCI 总线主板插槽的体积比原 ISA 总线插槽还小，支持突发读写操作，最大传输速率可达 132MB/s，可同时支持多组外围设备。PCI 局部总线不能兼容现有的 ISA、EISA、MCA（Micro Channel Architecture）总线，但它不受制于处理器，是基于奔腾等新一代微处理器而发展的总线。

SCSI（Small Computer System Interface）是一种用于计算机和智能设备之间（硬盘、软驱、光驱、打印机、扫描仪等）系统级接口的独立处理器标准。

参考答案

（6）C

试题（7）

下列安全协议中，与 TLS 最接近的协议是__（7）__。

（7）A．PGP　　B．SSL　　C．HTTPS　　D．IPSec

试题（7）分析

本题考查安全协议方面的基础知识。

SSL（Secure Socket Layer，安全套接层）是 Netscape 于 1994 年开发的传输层安全协议，用于实现 Web 安全通信。1996 年发布的 SSL 3.0 协议草案已经成为一个事实上的 Web 安全标准。

TLS（Transport Layer Security，传输层安全协议）是 IETF 制定的协议，它建立在 SSL 3.0 协议规范之上，是 SSL 3.0 的后续版本。

参考答案

（7）B

试题（8）、（9）

用户 B 收到用户 A 带数字签名的消息 M，为了验证 M 的真实性，首先需要从 CA

获取用户 A 的数字证书，并利用__（8）__验证该证书的真伪，然后利用__（9）__验证 M 的真实性。

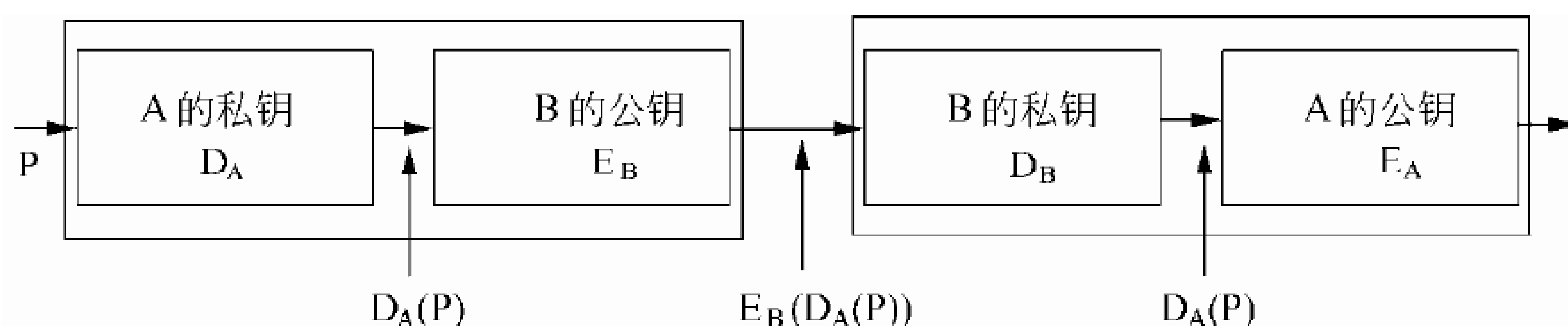
（8）A. CA 的公钥　B. B 的私钥　C. A 的公钥　D. B 的公钥

（9）A. CA 的公钥　B. B 的私钥　C. A 的公钥　D. B 的公钥

试题（8）、（9）分析

本题考查数字签名和数字证书方面的知识。

基于公钥的数字签名系统如下图所示：A 为了向 B 发送消息 P，A 用自己的私钥对 P 签名后再用 B 的公钥对签名后的数据加密，B 收到消息后先用 B 的私钥解密后在用 A 的公钥认证 A 的签名以及消息的真伪。



用户 B 收到用户 A 带数字签名的消息 M，为了验证 M 的真实性，首先需要从 CA 获取用户 A 的数字证书，验证证书的真伪需要用 CA 的公钥验证 CA 的签名，验证 M 的真实性需要用用户 A 的公钥验证用户 A 的签名。

参考答案

（8）A　（9）C

试题（10）

M 软件公司的软件产品注册商标为 M，为确保公司在市场竞争中占据优势，对员工进行了保密约束。此情形下该公司不享有__（10）__。

（10）A. 商业秘密权　B. 著作权　C. 专利权　D. 商标权

试题（10）分析

本题考查知识产权基础知识。

关于软件著作权的取得，《计算机软件保护条例》规定："软件著作权自软件开发完成之日起产生。"即软件著作权自软件开发完成之日起自动产生，不论整体还是局部，只要具备了软件的属性即产生软件著作权，既不要求履行任何形式的登记或注册手续，也无须在复制件上加注著作权标记，也不论其是否已经发表都依法享有软件著作权。软件开发经常是一项系统工程，一个软件可能会有很多模块，而每一个模块能够独立完成某一项功能。自该模块开发完成后就产生了著作权。软件公司享有商业秘密权。因为一项商业秘密受到法律保护的依据，必须具备构成商业秘密的三个条件，即不为公众所知悉、具有实用性、采取了保密措施。商业秘密权保护软件是以软件中是否包含着"商业秘密"为必要条件的。该软件公司组织开发的应用软件具有商业秘密的特征，即包含着他人不能知道到的技术秘密；具有实用性，能为软件公司带来经济效益；对职工进行了保密的

约束，在客观上已经采取相应的保密措施。所以软件公司享有商业秘密权。商标权、专利权不能自动取得，申请人必须履行商标法、专利法规定的申请手续，向国家行政部门提交必要的申请文件，申请获准后即可取得相应权利。获准注册的商标通常称为注册商标。

参考答案

（10）C

试题（11）

X软件公司的软件工程师张某兼职于Y科技公司，为完成Y科技公司交给的工作，做出了一项涉及计算机程序的发明。张某认为该发明是利用自己的业余时间完成的，可以以个人名义申请专利。此项专利申请权应归属 （11） 。

（11）A. 张某　　B. X软件公司　　C. Y科技公司　　D. 张某和Y科技公司

试题（11）分析

本题考查考生知识产权方面的基础知识。

专利法意义上的发明人必须是：第一，直接参加发明创造活动。在发明创造过程中，只负责组织管理工作或者是对物质条件的利用提供方便的人，不应当被认为是发明人；第二，必须是对发明创造的实质性特点作出创造性贡献的人。仅仅提出发明所要解决的问题而未对如何解决该问题提出具体意见的，或者仅仅从事辅助工作的人，不视为发明人或者设计人。有了发明创造不一定就能成为专利权人。发明人或设计人是否能够就其技术成果申请专利，还取决于该发明创造与其职务工作的关系。一项发明创造若被认定为职务发明创造，那么该项发明创造申请并获得专利的权利为该发明人或者设计人所属单位所有。根据专利法规定，职务发明创造分为两种情形：一是执行本单位的任务所完成的发明创造，二是主要是利用本单位的物质技术条件所完成的发明创造。《专利法实施细则》对“执行本单位的任务所完成的发明创造”和“本单位的物质技术条件”又分别作出了解释。所谓执行本单位的任务所完成的发明创造是指：①在本职工作中作出的发明创造；②履行本单位交付的本职工作之外的任务所作出的发明创造；③退职、退休或者调动工作后一年内所作出的，与其在原单位承担的本职工作或原单位分配的任务有关的发明创造。职务发明创造的专利申请权属于发明人所在的单位，但发明人或者设计人仍依法享有发明人身份权和获得奖励报酬的权利。

参考答案

（11）C

试题（12）

以下图像文件格式中， （12） 可以在单个文件中实现多个图片轮流显示或者构成简单的动画效果。

（12）A. BMP　　B. JPEG　　C. GIF　　D. TIFF

试题（12）分析

本题考查多媒体方面的基础知识。

BMP 图像文件格式是 Windows 操作系统采用的图像文件格式，BMP 采用位映射存储格式，除了图像深度可选以外，一般不采用其他任何压缩，所以占用的存储空间较大。

JPEG 图像文件格式是一种有损压缩的静态图像文件存储格式，压缩比例可以选择，支持灰度图像、RGB 真彩色图像和 CMYK 真彩色图像。JPG 格式文件的压缩比例很高，非常适用于处理大量图像的场合。

TIFF 是电子出版 CD-ROM 中的一个图像文件格式。TIFF 格式非常灵活易变，它又定义了 4 类不同的格式：TIFF-B 适用于二值图像；TIFF-G 适用于黑白灰度图像；TIFF-P 适用于带调色板的彩色图像；TIFF-R 适用于 RGB 真彩图像。无论在视觉上还是其他方面，都能把任何图像编码成二进制形式而不丢失任何属性。

GIF 图像文件格式以数据块为单位来存储图像的相关信息，采用了 LZW 无损压缩算法按扫描行压缩图像数据。它可以在一个文件中存放多幅彩色图像，每一幅图像都由一个图像描述符、可选的局部彩色表和图像数据组成。如果把存储于一个文件中的多幅图像逐幅读出来显示到屏幕上，可以像播放幻灯片那样显示或者构成简单的动画效果。GIF 定义了两种数据存储方式，一种是按行连续存储，存储顺序与显示器的显示顺序相同；另一种是按交叉方式存储，由于显示图像需要较长的时间，使用这种方法存放图像数据，用户可以在图像数据全部收到之前这幅图像的全貌，而不觉得等待时间太长。

采用图像文件格式支持实现的显示效果与通过软件的实现方式相比，循环显示图像的数量、每幅图像显示的时间、透明度变化等参数受图像文件格式的限制，而通过软件方式实现显示效果的各项参数则可以灵活配置。

参考答案

（12）C

试题（13）

计算机终端用户因等待时间过长而放弃下载或显示图片/影像的问题，不能通过采用<u>（13）</u>来解决。

（13）A．流媒体技术　　　　　B．JPEG 累进编码方式
　　　C．GIF 图像文件格式　　D．AD/DA 变换方式

试题（13）分析

本题考查多媒体基础知识。流媒体是指在网络中使用流式传输技术的连续时基媒体，而流媒体技术是指把连续的影像和声音信息经过压缩处理之后放到专用的流服务器上，让浏览者一边下载一边观看、收听，而不需要等到整个多媒体文件下载完成就可以即时观看和收听的技术。流媒体融合了多种网络以及音视频技术，在网络中要实现流媒体技术，必须完成流媒体的制作、发布、传播、播放等环节。

JPEG 累进（或增量、渐进、递增、progressive）编码模式，这样可以实现图像内容的方式传输，在浏览器上的直观效果是无须过久等待即可看到模糊的图像，然后图像显示内容由模糊逐渐变得清晰。

GIF 图像文件格式以数据块为单位来存储图像的相关信息，采用了 LZW 无损压缩算法按扫描行压缩图像数据。它可以在一个文件中存放多幅彩色图像，每一幅图像都由一个图像描述符、可选的局部彩色表和图像数据组成。如果把存储于一个文件中的多幅图像逐幅读出来显示到屏幕上，可以像播放幻灯片那样显示或者构成简单的动画效果。GIF 定义了两种数据存储方式，一种是按行连续存储，存储顺序与显示器的显示顺序相同；另一种是按交叉方式存储，由于显示图像需要较长的时间，使用这种方法存放图像数据，用户可以在图像数据全部收到之前这幅图像的全貌，而不觉得等待时间太长。

图像信号是一种模拟信号，计算机要对它进行处理，必须将它转换成为数字图像信号，即用二进制数字的编码形式来表示图像，转换过程一般称为 A/D 转换（模数转换）。将图像播放出来，需进行 D/A 转换（数模转换）。

参考答案

（13）D

试题（14）

计算机处理模拟视频信号过程中首先要进行__（14）__。

（14）A. A/D 变换　B. 数据压缩　C. D/A 变换　D. 数据存储

试题（14）分析

本题考查多媒体基础知识。视频信息是指活动的、连续的图像序列。一幅图像称为一帧，帧是构成视频信息的基本单元。全屏幕视频是指显示的视频图像充满整个屏幕，能以 30 帧/秒的速度刷新画面，使画面不会产生闪烁和不连贯的现象。电视机、激光视盘、摄像机等都可提供丰富多彩的模拟视频信号，常常需要把这些信号与计算机图形图像结合在一个共同的空间，通过处理达到最佳的效果，然后输出到计算机的显示器或其他电视设备上。模拟视频信号进入计算机，首先需要解决模拟视频信息的数字化问题。视频数字化的目的是将模拟信号经模数转换和彩色空间变换等过程，转换成计算机可以显示和处理的数字信号。由于电视和计算机的显示机制不同，因此要在计算机上显示视频图像需要作许多处理。例如，电视是隔行扫描，计算机的显示器通常是逐行扫描；电视是亮度（Y）和色度（C）的复合编码，而 PC 机的显示器工作在 RGB 空间；电视图像的分辨率和显示屏的分辨率也各不相同等。这些问题在电视图像数字化过程中都需考虑。一般，对模拟视频信息进行数字化采取如下方式：

① 先从复合彩色电视图像中分离出彩色分量，然后数字化。目前市场上的大多数电视信号都是复合的全电视信号，如录像带、激光视盘等存储设备上的电视信号。对这类信号的数字化，通常是将其分离成 YUV、YIQ 或 RGB 彩色空间的分量信号，然后用 3 个 A/D 转换器分别进行数字化。这种方式称为复合数字化。

② 先对全彩色电视信号数字化，然后在数字域中进行分离，以获得 YUV、YIQ 或 RGB 分量信号。用这种方法对电视图像数字化时，只需一个高速 A/D 转换器。这种方式称为分量数字化。

分量数字化方式是较多使用的一种方式。电视信号使用的彩色空间是 YUV 空间，即每幅彩色画面有亮度（Y）和色度（U、V）3 个分量，对这 3 个分量需分别进行取样和量化，得到一幅数字图像。由于人眼对色度信号的敏感程度远不如对亮度信号那么灵敏，所以色度信号的取样频率可以比亮度信号的取样频率低一些，以减少数字视频的数据量。

参考答案

（14）A

试题（15）

以下关于数据流图的叙述中，不正确的是＿（15）＿。

（15）A．每条数据流的起点或终点必须是加工

B．必须保持父图与子图平衡

C．每个加工必须有输入数据流，但可以没有输出数据流

D．应保持数据守恒

试题（15）分析

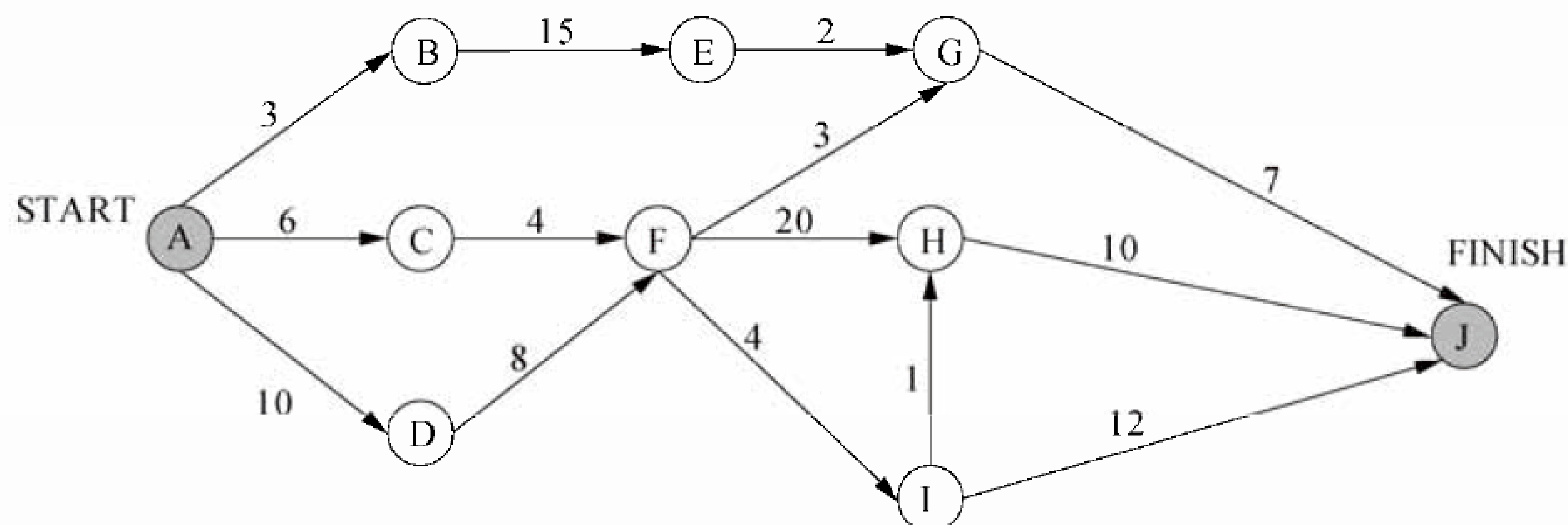
数据流图是结构化分析方法的重要模型，用于描述系统的功能、输入、输出和数据存储等。在绘制数据流图中，每条数据流的起点或者终点必须是加工，即至少有一端是加工。在分层数据流图中，必须要保持父图与子图平衡。每个加工必须既有输入数据流又有输出数据流。必须要保持数据守恒。也就是说，一个加工所有输出数据流中的数据必须能从该加工的输入数据流中直接获得，或者是通过该加工能产生的数据。

参考答案

（15）C

试题（16）、（17）

某软件项目的活动图如下所示。图中顶点表示项目里程碑，连接顶点的边表示包含的活动，则里程碑＿（16）＿在关键路径上，活动 FG 的松弛时间为＿（17）＿。



（16）A．B　　B．C　　C．D　　D．I

（17）A．19　　B．20　　C．21　　D．24

试题（16）、（17）分析

该活动图的关键路径为 ADFHJ，关键路径长度为 48 天，因此里程碑 D 在关键路径上，B、C 和 I 步骤关键路径上。活动 FG 的最早开始时间为第 19 天，最晚开始时间为第 39 天，因此松弛时间为 20 天。

参考答案

（16）C （17）B

试题（18）

在软件设计阶段，划分模块的原则是：一个模块的 （18） 。

（18）A．作用范围应该在其控制范围之内
　　　B．控制范围应该在其作用范围之内
　　　C．作用范围与控制范围互不包含
　　　D．作用范围与控制范围不受任何限制

试题（18）分析

模块的作用范围定义为受该模块内一个判定影响的模块集合，模块的控制范围为模块本身以及所有直接或间接从属于该模块的模块集合。其作用范围应该在控制范围之内。

参考答案

（18）A

试题（19）

定义风险参照水准是 （19） 活动常用的技术。

（19）A．风险识别　　B．风险预测　　C．风险评估　　D．风险控制

试题（19）分析

定义风险参照水准是风险评估的一类技术，对于大多数软件项目来说成本、速度和性能是三种典型的风险参照水准。

参考答案

（19）C

试题（20）

可用于编写独立程序和快速脚本的语言是 （20） 。

（20）A．Python　　B．Prolog　　C．Java　　D．C#

试题（20）分析

本题考查程序语言基础知识。

脚本语言又被称为扩建的语言，或者动态语言，是一种编程语言，通常以文本（如 ASCII）保存，只在被调用时进行解释或编译。Python 是一种脚本语言。

参考答案

（20）A

试题（21）

语言 $L=\{a^m b^n \mid m \geqslant 0, n \geqslant 1\}$ 的正规表达式是__（21）__。

（21）A. aa^*bb^*　　B. a^*bb^*　　C. aa^*b^*　　D. a^*b^*

试题（21）分析

本题考查程序语言基础知识。

aa^*bb^* 表示的字符串特点是：若干个 a 之后跟若干个 b，a 和 b 都至少出现 1 次。

a^*bb^* 表示的字符串特点是：若干个 a 之后跟若干个 b，a 可以不出现，b 至少出现 1 次。

aa^*b^* 表示的字符串特点是：若干个 a 之后跟若干个 b，a 至少出现 1 次，b 可以不出现。

a^*b^* 表示的字符串特点是：若干个 a 之后跟若干个 b，a 和 b 都可以不出现。

语言 $L=\{a^m b^n \mid m \geqslant 0, n \geqslant 1\}$ 中，若干个 a 之后跟若干个 b，a 可以不出现，b 至少出现 1 次。

参考答案

（21）B

试题（22）

算术表达式(a–b)*c+d 的后缀式是__（22）__（–、+、*表示算术的减、加、乘运算，运算符的优先级和结合性遵循惯例）。

（22）A. a b c d – * +　　B. a b – c d * +

C. a b – c * d +　　D. a b c – d * +

试题（22）分析

本题考查程序语言基础知识。

后缀式即逆波兰式，是逻辑学家卢卡西维奇发明的一种表示表达式的方法。这种表示方式把运算符写在运算对象的后面，例如，把 a+b 写成 ab+。这种表示法的优点是根据运算对象和算符的出现次序进行计算，不需要使用括号。

(a–b)*c+d 的后缀式是 ab-c*d+。

参考答案

（22）C

试题（23）

某系统中仅有 5 个并发进程竞争某类资源，且都需要 3 个该类资源，那么至少有__（23）__个该类资源，才能保证系统不会发生死锁。

（23）A. 9　　B. 10　　C. 11　　D. 15

试题（23）分析

本题考查操作系统进程管理方面的基础知识。

假设系统为每个进程分配了 2 个资源，对于选项 C，系统还剩余 1 个资源，能保证

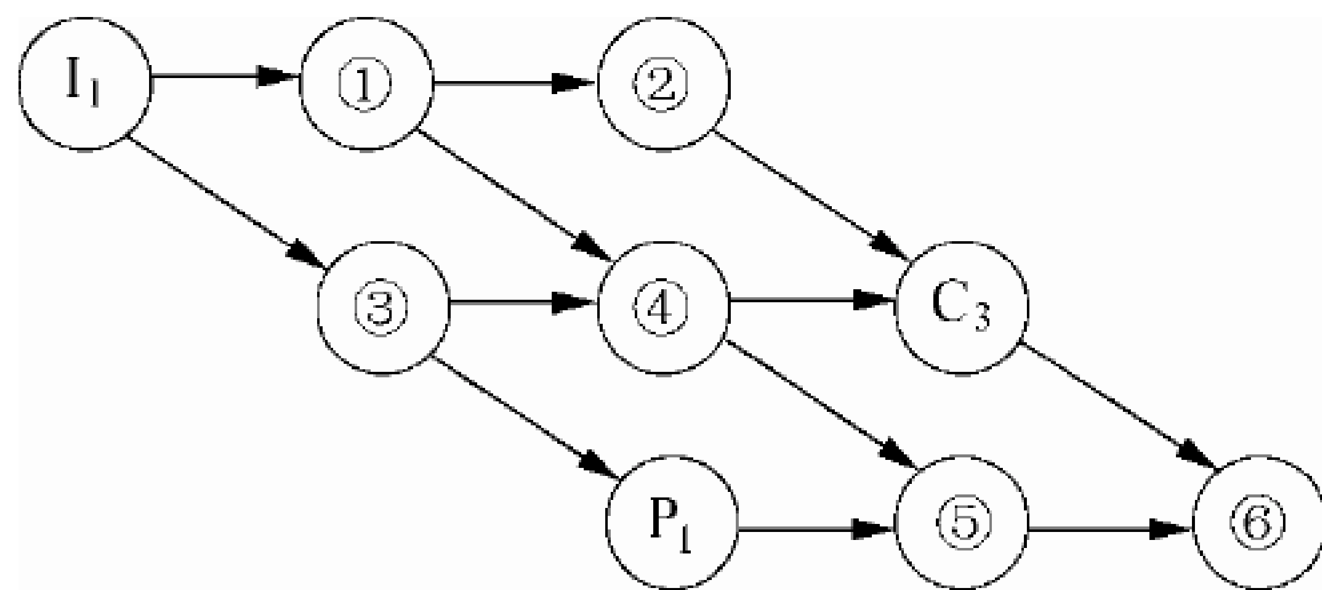
5个进程中的一个进程运行完毕。当该进程释放其占有的资源，系统可用资源数为3个，能保证未完成的4个进程中的3个进程运行完毕。当这3个进程释放其占有的资源，系统可用资源数为9个，显见能确保最后一个进程运行完。

参考答案

（23）C

试题（24）～（26）

某计算机系统中有一个CPU、一台输入设备和一台输出设备，假设系统中有三个作业T1、T2和T3，系统采用优先级调度，且T1的优先级>T2的优先级>T3的优先级。若每个作业具有三个程序段：输入I_i、计算C_i和输出P_i（i = 1,2,3），执行顺序为I_i、C_i、P_i，则这三个作业各程序段并发执行的前驱图如下所示。图中①、②分别为__（24）__，③、④分别为__（25）__，⑤、⑥分别为__（26）__。



（24）A. I_2、C_2　　B. I_2、I_3　　C. C_1、P_3　　D. C_2、P_2

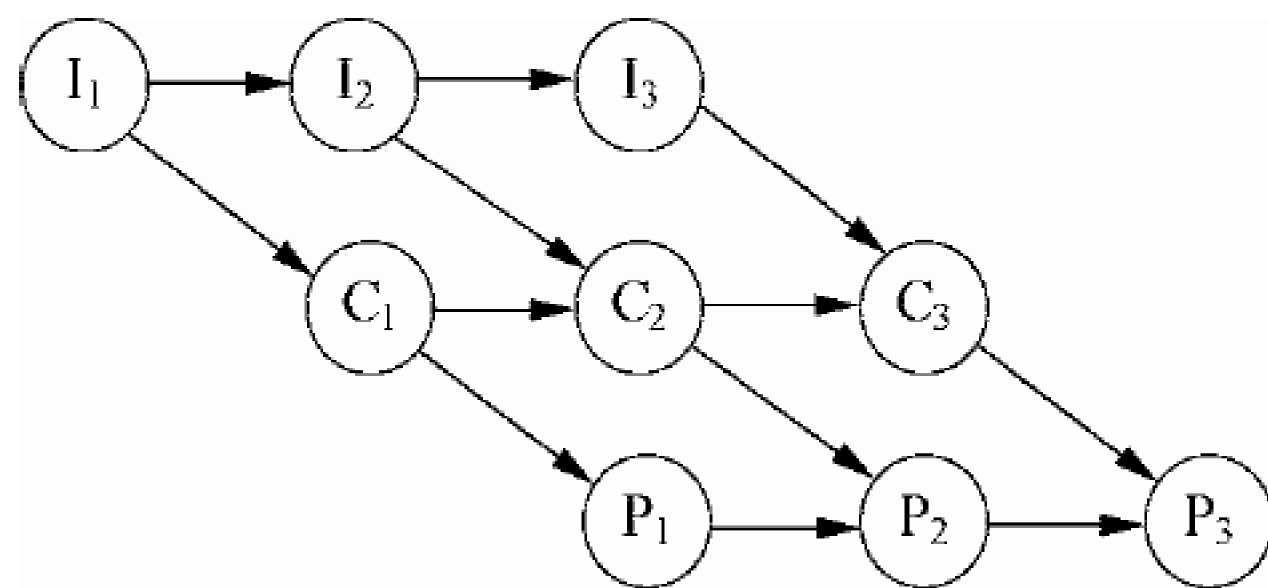
（25）A. C_1、C_2　　B. I_2、C_1　　C. I_3、P_3　　D. C_1、P_2

（26）A. I_3、C_2　　B. I_2、C_1　　C. P_2、P_3　　D. C_1、P_2

试题（24）～（26）分析

本题考查操作系统前驱图方面的基础知识。

前趋图是一个有向无循环图，由节点和有向边组成，节点代表各程序段的操作，而节点间的有向边表示两个程序段操作之间存在的前趋关系（“→”）。程序段Pi和Pj的前趋关系可表示成Pi→Pj，其中Pi是Pj的前趋，Pj是Pi的后继，其含义是Pi执行结束后Pj才能执行。本题完整的前趋图如下图所示。



根据题意，I_1执行结束后C_1才能执行，C_1执行结束后P_1才能执行，因此I_1是C_1、P_1的前趋，C_1是P_1的前趋。可见，图中③应为C_1。又因为计算机系统中只有一台输入设备，所以I_1执行结束后I_2和I_3才能执行，故I_1是I_2和I_3的前趋， I_2是I_3的前趋。可见，图中①、②分别为I_2、I_3。

经上分析③应为C_1，所以C_1是P_1的前趋。又因为计算机系统中只有一个CPU，而

且系统采用优先级调度，所以 C_1 是 C_2 的前趋，C_2 是 C_3 的前趋。可见，图中④应为 C_2。

因为计算机系统中只有一台输出设备，所以 P_1 执行结束后 P_2 才能执行，故 P_1 是 P_2 的前趋；P_2 执行结束后 P_3 才能执行，P_2 是 P_3 的前趋。可见，图中⑤、⑥分别为 P_2、P_3。

参考答案

（24）B （25）A （26）C

试题（27）、（28）

设文件索引节点中有 8 个地址项，每个地址项大小为 4 字节，其中 5 个地址项为直接地址索引，2 个地址项是一级间接地址索引，1 个地址项是二级间接地址索引，磁盘索引块和磁盘数据块大小均为 1KB。若要访问文件的逻辑块号分别为 5 和 518，则系统应分别采用 （27） ；而且可表示的单个文件最大长度是 （28） KB。

（27）A．直接地址索引和一级间接地址索引
B．直接地址索引和二级间接地址索引
C．一级间接地址索引和二级间接地址索引
D．一级间接地址索引和一级间接地址索引

（28）A．517 B．1029 C．16513 D．66053

试题（27）、（28）分析

本题考查操作系统文件管理方面的基础知识。

根据题意，磁盘索引块为 1KB 字节，每个地址项大小为 4 字节，故每个磁盘索引块可存放 1024/4=256 个物理块地址。又因为文件索引节点中有 8 个地址项，其中 5 个地址项为直接地址索引，这意味着逻辑块号为 0～4 的为直接地址索引；2 个地址项是一级间接地址索引，这意味着第一个地址项指出的物理块中存放逻辑块号为 5～260 的物理块号，第一个地址项指出的物理块中存放逻辑块号为 261～516 的物理块号；1 个地址项是二级间接地址索引，该地址项指出的物理块存放了 256 个间接索引表的地址，这 256 个间接索引表存放逻辑块号为 517～66052 的物理块号。

经上分析不难得出，若要访问文件的逻辑块号分别为 5 和 518，则系统应分别采用一级间接地址索引和二级间接地址索引。

单个文件的逻辑块号可以从 0～66052，而磁盘数据块大小为 1KB 字节，所以单个文件最大长度是 66053KB。

参考答案

（27）C （28）D

试题（29）

某开发小组欲开发一个规模较大、需求较明确的项目。开发小组对项目领域熟悉且该项目与小组开发过的某一项目相似，则适宜采用 （29） 开发过程模型。

（29）A．瀑布 B．演化 C．螺旋 D．喷泉

试题（29）分析

项目规模大、开发小组对项目需求理解并了解相关领域，因此可以采用瀑布开发模型。演化模式适用于对软件需求缺乏准确认识的情况。螺旋模型在开发过程中加入风险

分析。喷泉模型适合于面向对象的开发方法。

参考答案

（29）A

试题（30）

敏捷开发方法中，__（30）__认为每一种不同的项目都需要一套不同的策略、约定和方法论。

（30）A．极限编程（XP） B．水晶法（Crystal）
C．并列争球法（Scrum） D．自适应软件开发（ASD）

试题（30）分析

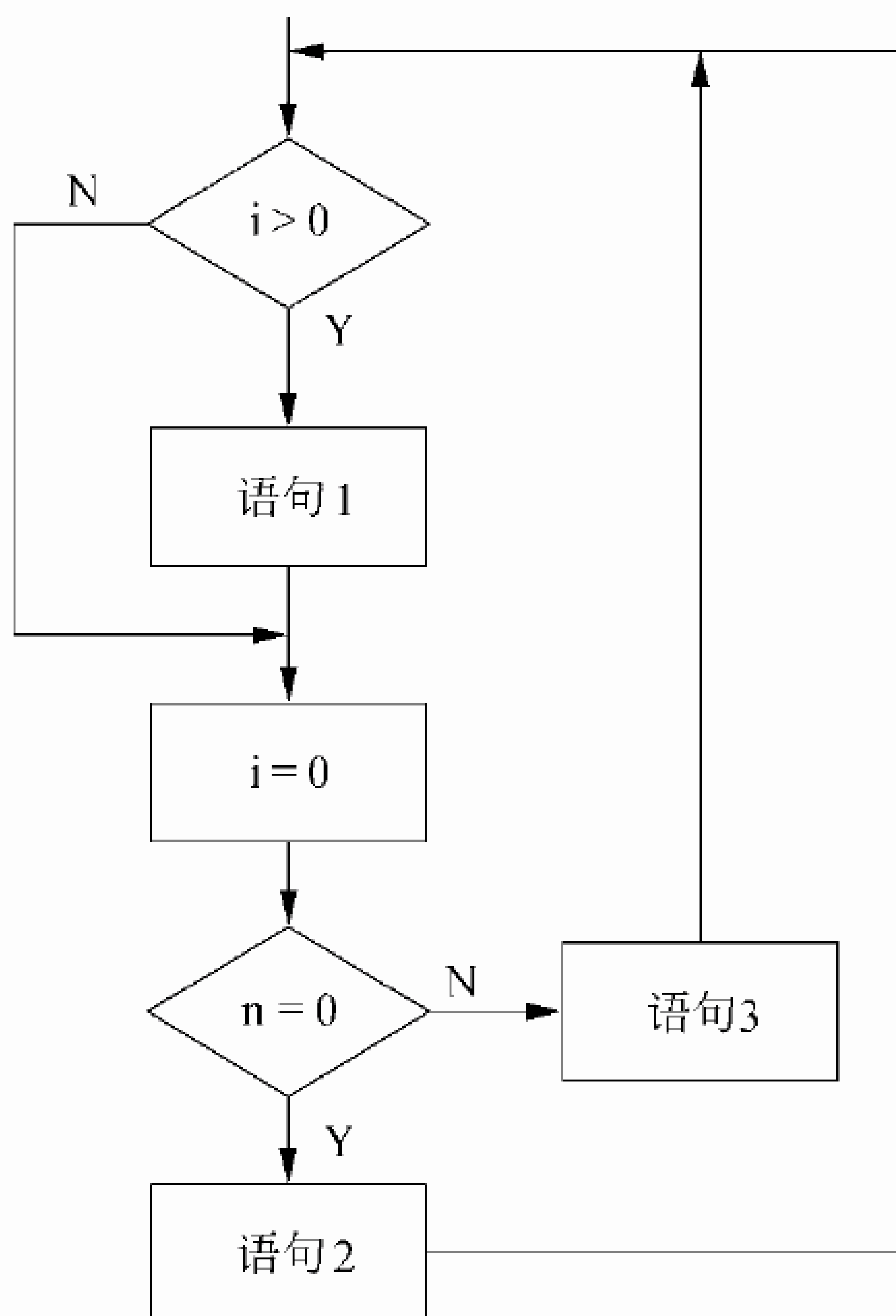
敏捷开发的总体目标是通过“尽可能早地、持续地对有价值的软件的交付”使客户满意。敏捷过程的典型方法很多，每一种方法基于一套原则，这些原则实现了敏捷方法所宣称的理念，即敏捷宣言。其中，极限编程 XP 是一种轻量级的软件开发方式，由价值观、原则、实践和行为 4 个部分组成，彼此相互依赖、关联，并通过行为贯穿于整个生存周期。水晶法 Crystal 认为每一个不同的项目都需要一套不同的策略、约定和方法论。并列争球法 Scrum 使用迭代的方法，并按需求的优先级来实现产品。自适应软件开发 ASD 有 6 个基本原则。

参考答案

（30）B

试题（31）

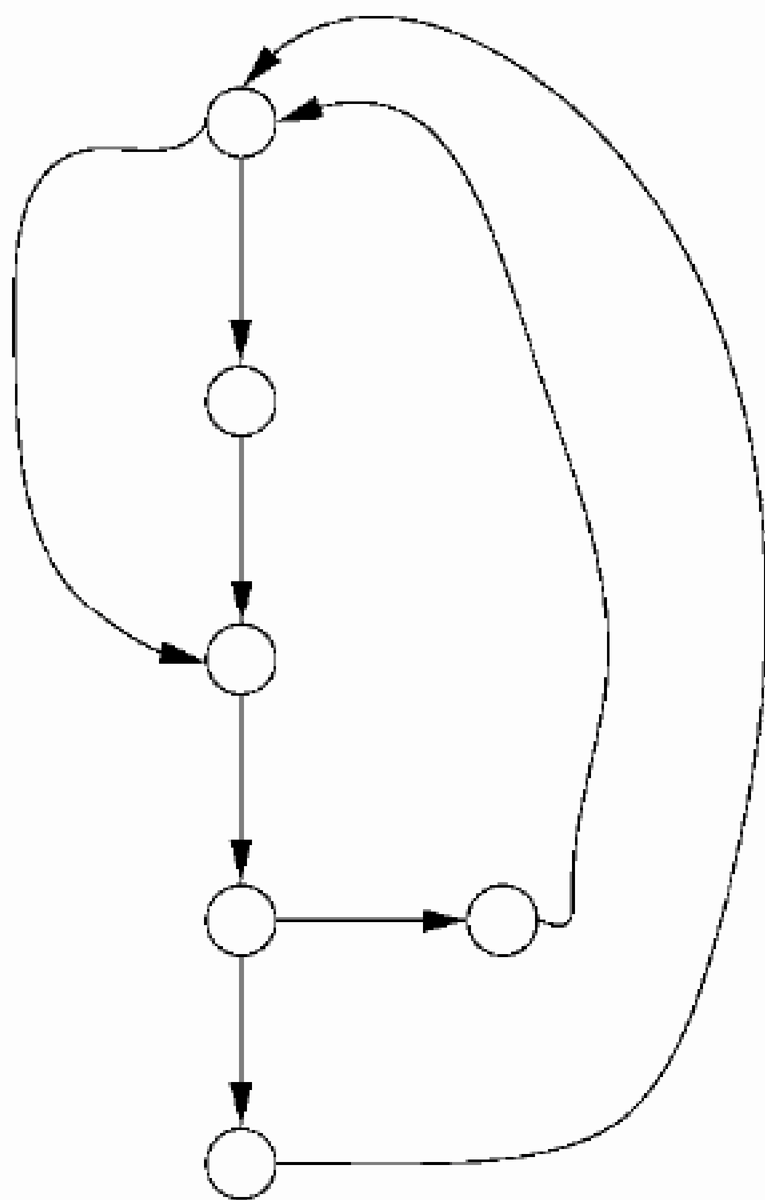
采用 McCabe 度量法计算下图的环路复杂性为__（31）__。



（31）A．2　　B．3　　C．4　　D．5

试题（31）分析

题图可以用下图表示，图中顶点数位 6，边数为 8，程序复杂度为 m–n+2=8 – 6 + 2 = 4。



参考答案

（31）C

试题（32）

在屏蔽软件错误的容错系统中，冗余附加技术的构成不包括__（32）__。

（32）A．关键程序和数据的冗余存储及调用

B．冗余备份程序的存储及调用

C．实现错误检测和错误恢复的程序

D．实现容错软件所需的固化程序

试题（32）分析

冗余附加技术是指为实现结构、信息和时间冗余技术所需的资源和技术，包括程序、指令、数据、存放和调动它们的空间和通道等。在屏蔽硬件错误的容错技术中，冗余附加技术包括：关键程序和数据的冗余及调用；检测、表决、切换、重构和复算的实现。在屏蔽软件错误的容错技术中，冗余附加技术包括：冗余备份程序的存储及调用；实现错误检测和错误恢复的程序；实现容错软件所需的固化程序。

参考答案

（32）A

试题（33）

以下关于文档的叙述中，不正确的是__（33）__。

（33）A．文档仅仅描述和规定了软件的使用范围及相关的操作命令

B．文档是软件产品的一部分，没有文档的软件不能称之为软件产品

C．软件文档的编制在软件开发工作中占有突出的地位和相当大的工作量

D．高质量文档对于发挥软件产品的效益有着重要的意义

试题（33）分析

文档是系统建设过程的“痕迹”，是系统维护人员的指南，是开发人员与用户交流的工具。文档不仅仅描述和规定软件的适用范围及相关的操作命令。软件包括程序和文档，因此没有文档的软件不能称之为软件产品。软件文档的编制在软件开发中是相当重要的，高质量的文档对于发挥软件产品的效益有着重要的意义。

参考答案

（33）A

试题（34）

由于信用卡公司升级了其信用卡支付系统，导致超市的原有信息系统也需要做相应的修改工作，该类维护属于<u>（34）</u>。

（34）A．正确性维护　　B．适应性维护
C．完善性维护　　D．预防性维护

试题（34）分析

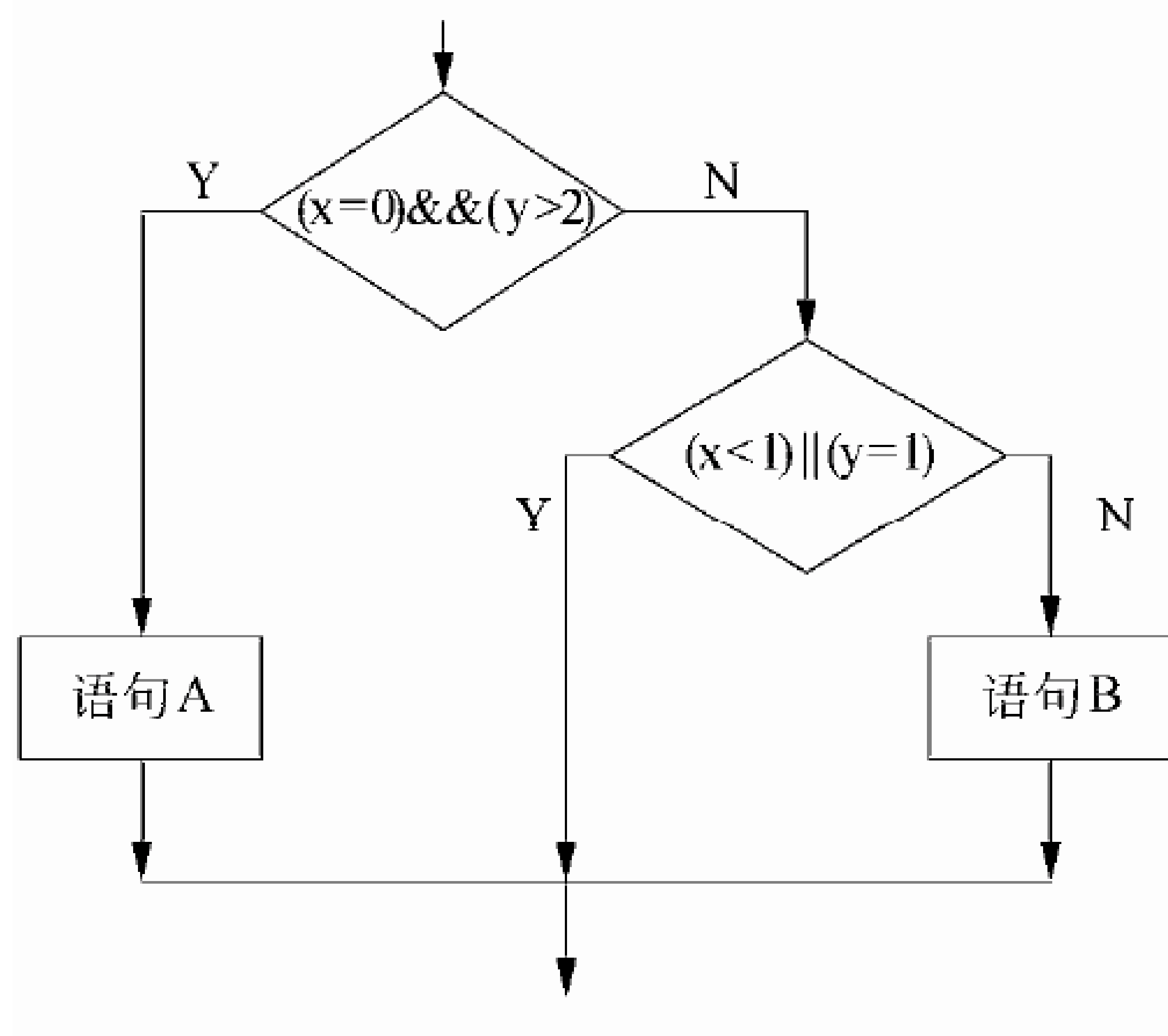
软件维护一般包括下面四个方面。正确性维护是指改正在系统开发阶段已经发生而在系统测试阶段尚未发生的错误。适应性维护是指使应用软件适应信息技术变化和管理需求变化而进行的修改。完善性维护为扩充功能和改善性能而进行的修改。预防性维护是为了改进应用软件的可靠性和可维护性，为了适应未来的软硬件环境的编号，主动增加预防性的新的功能，以使应用系统适应各类变化而不被淘汰。本题超市信息系统为了适应信用卡支付系统而做了相应的修改工作，是一个典型的适应性维护。

参考答案

（34）B

试题（35）、（36）

用白盒测试方法对下图所示的程序进行测试，设计了 4 个测试用例：①（x = 0, y=3）、②（x = 1, y = 2）、 ③（x = –1, y = 2）和④（x = 3, y = 1）。测试用例①②实现了<u>（35）</u>覆盖；若要完成路径覆盖，则可用测试用例<u>（36）</u>。



（35）A．语句　B．条件　C．判定　D．路径

（36）A．①②　B．②③　C．①②③　D．①③④

试题（35）、（36）分析

白盒测试也称为结构测试，根据程序的内部结构和逻辑来设计测试用例，对程序的路径和过程进行测试，检查是否满足设计的需要。其常用的技术有逻辑覆盖、循环覆盖和基本路径测试。

在逻辑覆盖中，语句覆盖是指选择足够的测试数据使被测试程序中每条语句至少执行一次。判定覆盖是指选择足够的测试数据使被测试程序中每个判定表达式至少获得一次“真”值和“假”值。条件覆盖是指构造一组测试用例，使得每一判定语句中每个逻辑条件的各种可能的值至少满足一次。路径覆盖是指覆盖被测程序中所有可能的路径。

本题的实例中，测试用例①会执行语句 A，测试用例②会执行语句 B，测试用例③和④不执行语句。因此测试用例①②可以完成语句覆盖，不能完成判定、条件和路径覆盖。要完成路径覆盖，需要测试用例①②③或测试用例①②④。

参考答案

（35）A　（36）C

试题（37）

在面向对象技术中，对象具有以下特性：__（37）__。

① 清晰的边界　② 良好定义的行为　③ 确定的位置和数量　④ 可扩展性

（37）A．②④　B．①②③④　C．①②④　D．①②

试题（37）分析

本题考查面向对象的基本知识。

在面向对象技术中，对象是基本的运行时的实体，它既包括数据（属性），也包括作用于数据的操作（行为）。一个对象把属性和行为封装为一个整体，与其他对象之间有清晰的边界，有良好定义的行为和可扩展性。对象位置和数量，由使用其的对象或系统确定。

参考答案

（37）C

试题（38）、（39）

在面向对象技术中，__（38）__说明一个对象具有多种形态，__（39）__定义超类与子类之间的关系。

（38）A．继承　B．组合　C．封装　D．多态

（39）A．继承　B．组合　C．封装　D．多态

试题（38）、（39）分析

本题考查面向对象的基本知识。

面向对象技术中，继承关系是一种模仿现实世界中继承关系的一种类之间的关系，

是超类（父类）和子类之间共享数据和方法的机制。在定义和实现一个类的时候，可以在一个已经存在的类的基础上来进行，把这个已经存在的类所定义的内容作为自己的内容，并加入新的内容。组合表示对象之间整体与部分的关系。封装是一种信息隐藏技术，其目的是使对象（组件）的使用者和生产者分离，也就是使其他开发人员无须了解所要使用的软件组件内部的工作机制，只需知道如何使用组件，即组件提供的功能及其接口。多态（polymorphism）是不同的对象收到同一消息可以产生完全不同的结果的现象，使得用户可以发送一个通用的消息，而实现的细节则由接收对象自行决定，达到同一消息就可以调用不同的方法，即多种形态。

参考答案

（38）D　　（39）A

试题（40）

如果要表示待开发软件系统中软件组件和硬件之间的物理关系，通常采用 UML 中的＿（40）＿。

（40）A．组件图　　B．部署图　　C．类图　　D．网络图

试题（40）分析

本题考查统一建模语言（UML）的基本知识。

UML2.0 中提供了多种图形。组件图（component diagram）展现了一组组件之间的组织和依赖，专注于系统的静态实现视图，与类图相关，通常把组件映射为一个或多个类、接口或协作。部署图（deployment diagram）展现了运行处理节点以及其中构件的配置。部署图给出了体系结构的静态实施视图。它与构件图相关，通常一个节点包含一个或多个构件。类图（class diagram）展现了一组对象、接口、协作和它们之间的关系，在开发软件系统时，类图用于对系统的静态设计视图建模。

参考答案

（40）B

试题（41）

对于场景：一个公司负责多个项目，每个项目（Project）由一个员工（Employee）团队（Team）来开发。下列 UML 概念图中，＿（41）＿最适合描述这一场景。

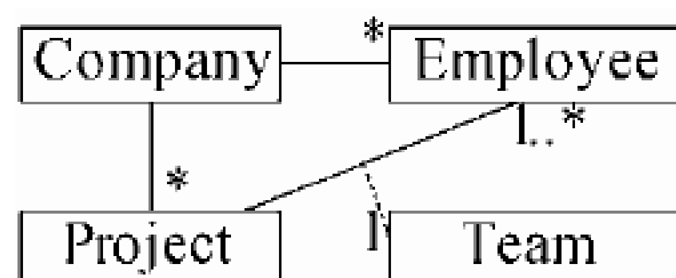


图 A

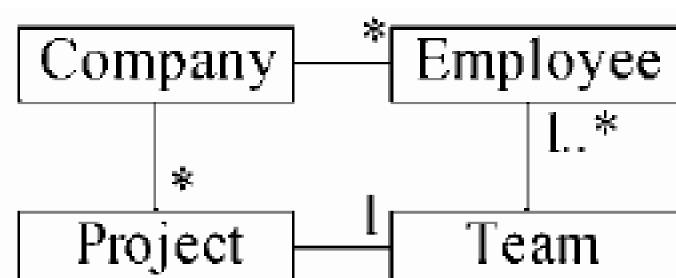


图 B

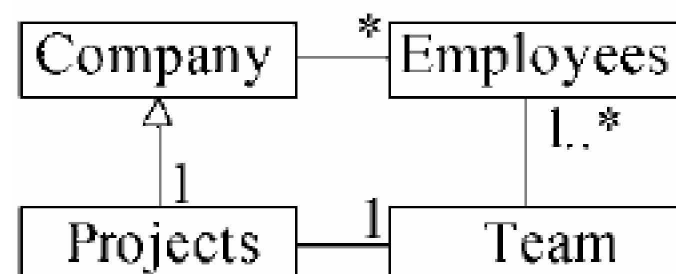


图 C

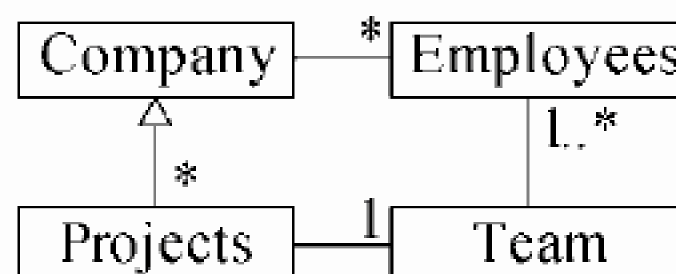


图 D

（41）A．图 A　　　B．图 B　　　C．图 C　　　D．图 D

试题（41）分析

本题考查采用统一建模语言（UML）进行面向对象建模的基本知识。

面向对象分析是从按对象分类的角度来创建对象领域的描述。领域的分解包括定义概念、属性和重要的关联。其结果可以被表示成领域模型，用一组显示领域概念或对象的图形来表示领域模型。通常首先发现和确定业务对象，然后组织对象并记录对象之间的主要概念关系。类图以图形化的用以描述对象及其关联关系。在该图中还包括多重性、关联关系、泛化/特化关系以及聚合关系。

从场景中先识别名词性术语，包括公司（Company）、项目（Project）、员工（Employee）、和团队（Team），再识别这些术语之间的关联关系。一个公司负责多个项目，公司和项目之间具有 1 对多的关联关系；一个项目由一个员工团队来开发，项目到团队的管理关系是 1，而一个（员工）团队是由多名员工组成，而且没有员工称不上是团队，所以一个团队至少和一个员工关联。

参考答案

（41）B

试题（42）

UML 中接口可用于＿（42）＿。

（42）A．提供构造型（stereotype）名称为<<interface>>的具体类

B．Java 和 C++程序设计中，而 C#程序设计中不支持

C．定义可以在多个类中重用的可执行逻辑

D．声明对象类所需要的服务

试题（42）分析

本题考查统一建模语言（UML）建模的基本知识。

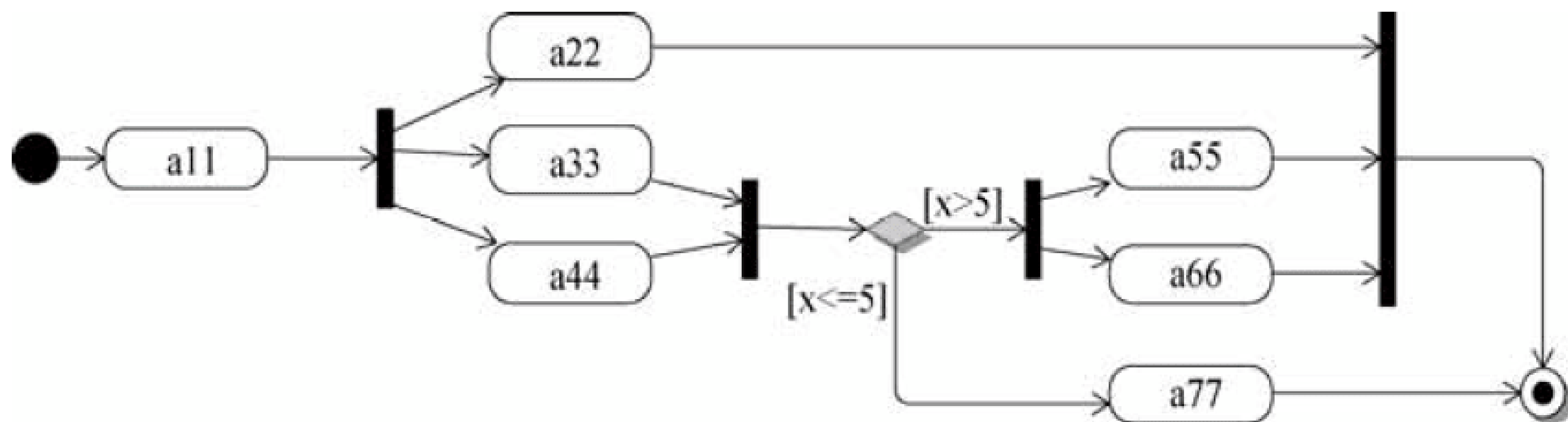
在采用 UML 进行面向对象系统建模时，会用 UML 中构造型（stereotype）名称为<<interface>>来表示接口这一概念，声明对象类所需要的服务，而服务具体如何执行，有实现它的具体类完成。

参考答案

（42）D

试题（43）

下列活动图中可以同时执行的活动是＿（43）＿。



（43）A. a44 和 a66　　B. a22，a33 和 a44
C. a11 和 a77　　D. a66 和 a77

试题（43）分析

本题考查统一建模语言（UML）的基本知识。

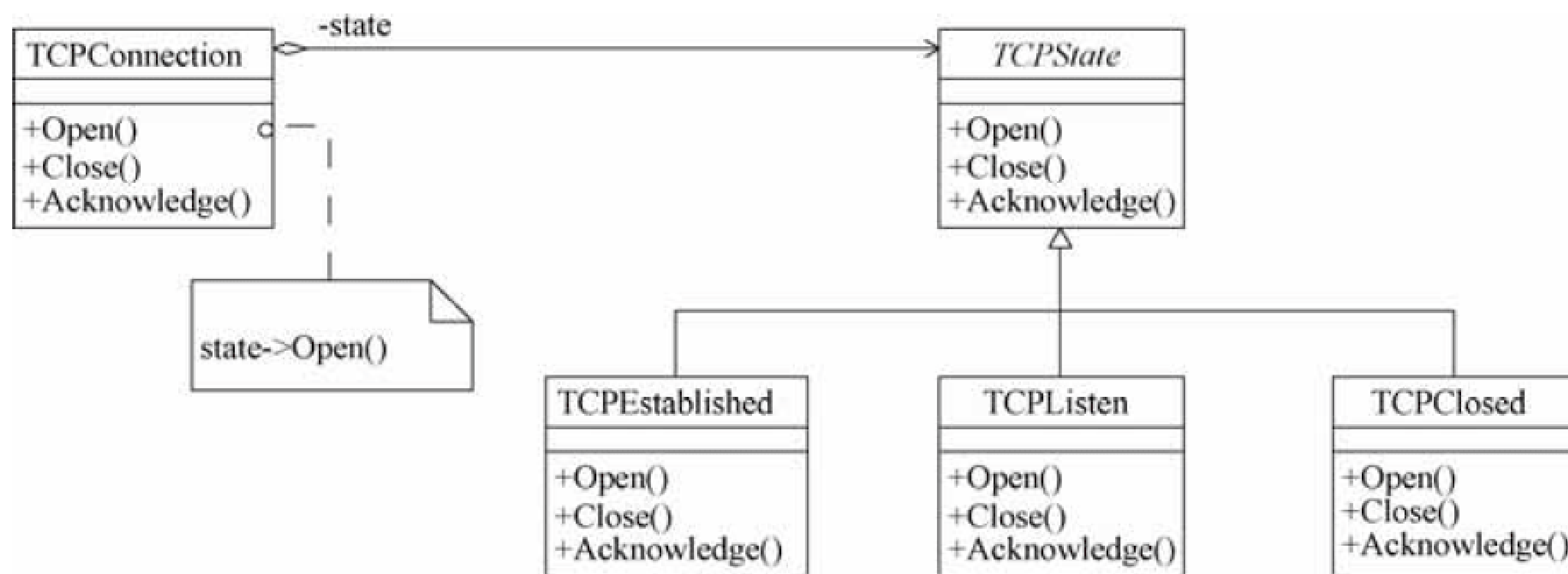
UML 2.0 中提供了的活动图（activity diagram）是一种特殊的状态图，它展现了在系统内从一个活动到另一个活动的流程。活动图专注于系统的动态视图，它对于系统的功能建模特别重要，并强调对象间的控制流程。通常用于在建模用例图之后，对复杂用例进行进一步细化。活动图中可以用条状图表示同步的起始点和结束点，其间的活动可以同时执行，如图中 a22、a33 和 a44 在 a11 执行完后，到同步起始，其后各自执行，同步结束后的活动必须等同步结束点之前的活动全部执行完之后才能继续，如 a33 和 a44 都结束后，才进入后续判定。

参考答案

（43）B

试题（44）～（46）

每种设计模式都有特定的意图。__（44）__模式使得一个对象在其内部状态改变时通过调用另一个类中的方法改变其行为，使这个对象看起来如同修改了它的类。下图是采用该模式的有关 TCP 连接的结构图实例。该模式的核心思想是引入抽象类__（45）__来表示 TCP 连接的状态，声明不同操作状态的公共接口，其子类实现与特定状态相关的行为。当一个__（46）__对象收到其他对象的请求时，它根据自身的当前状态做出不同的反应。



（44）A. 适配器（Adapter）　　B. 命令（Command）
C. 观察者（Visitor）　　D. 状态（State）

（45）A. TCPConnection　　B. state
C. TCPState　　D. TCPEstablished

（46）A. TCPConnection　　B. state
C. TCPState　　D. TCPEstablished

试题（44）～（46）分析

本题考查设计模式的基本知识。

每种设计模式都有特定的意图。适配器（Adapter）模式将一个类的接口转换成客户希望的另外一个接口，使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。命令（Command）模式将一个请求封装为一个对象，从而使使用者可以采用不同的请求对客户进行参数化；对请求排队或记录请求日志，以及支持可撤销的操作。观察者（Visitor）模式表示一个作用于某对象结构中的各元素的操作，是使用者可以在不改变各元素的类的前提下定义作用于这些元素的新操作。状态（State）模式是使得一个对象在其内部状态改变时通过调用另一个类中的方法改变其行为，使这个对象看起来如同修改了它的类。

题目中是一个标识网络连接的实例，网络连接类为 TCPConnection，其对象的状态处于不同状态之一：连接已建立（Established）、正在监听（Listening）、连接已关闭（Closed）。当一个 TCPConnection 对象受到其他对象的请求时，它根据自身的当前状态作出相应的反应。例如，一个 Open 请求的结果依赖于该连接是出于连接已关闭状态还是连接已建立状态。State 模式描述了 TCPConnection 如何做每一种状态下表现出不同的行为。这一模式思想是引入了一个称为 TCPState 的抽象类来表示网络的连接状态。TCPState 类为各表示不同的操作状态的子类声明了一个公共接口。TCPState 的子类实现与特定状态相关的行为。例如，TCPEstablished 和 TCPClosed 类分别实现了特定于 TCPConnection 的连接已建立状态和连接已关闭状态的行为。

参考答案

（44）D　　（45）C　　（46）A

试题（47）

欲使类 A 的所有使用者都使用 A 的同一个实例，应__（47）__。

（47）A．将 A 标识为 final

B．将 A 标识为 abstract

C．将单例（Singleton）模式应用于 A

D．将备忘（Memento）模式应用于 A

试题（47）分析

本题考查设计模式的基本知识。

每一个设计模式描述了一个在我们周围不断重复发生的问题，以及该问题的解决方案的核心，使该方案能够重用而不必做重复劳动。

将类标识为 final 限制类不能再被继承；将设计为 abstract 表示类中定义出类提供什么服务，而有些具体服务需要通过其子类来实现；单例（Singleton）模式是指系统运行过程中，一个类只有一个对象实例；备忘模式是指不破坏封装性的前提下捕获一个对象的内部状态，并在该对象之外保存这个状态。

参考答案

（47）C

试题（48）

将高级语言源程序翻译成目标程序的是__（48）__。

（48）A．解释程序　B．编译程序　C．链接程序　D．汇编程序

试题（48）分析

本题考查程序语言基础知识。

计算机只能理解和执行由 0、1 序列构成的机器语言，因此高级程序语言需要翻译，担负这一任务的程序称为“语言处理程序”。由于应用的不同，语言之间的翻译也是多种多样的。语言处理程序主要分为汇编程序、编译程序和解释程序三种基本类型。

解释程序也称为解释器，它可以直接解释执行源程序，或者将源程序翻译成某种中间表示形式后再加以执行；而编译程序（编译器）则首先将源程序翻译成目标语言程序，然后在计算机上运行目标程序。汇编程序的功能是将汇编语言所编写的源程序翻译成机器指令程序。

链接程序将各目标程序连接形成为可执行程序。

参考答案

（48）B

试题（49）

在对程序语言进行翻译的过程中，常采用一些与之等价的中间代码表示形式。常用的中间代码表示不包括__（49）__。

（49）A．树　B．后缀式　C．四元式　D．正则式

试题（49）分析

本题考查程序语言基础知识。

从原理上讲，对源程序进行语义分析之后就可以直接生成目标代码，但由于源程序与目标代码的逻辑结构往往差别很大，特别是考虑到具体机器指令系统的特点，要使翻译一次到位很困难，而且用语法制导方式机械生成的目标代码往往是烦琐和低效的，因此有必要采用一种中间代码，将源程序首先翻译成中间代码表示形式，以利于进行与机器无关的优化处理。由于中间代码实际上也起着编译器前端和后端分水岭的作用，所以使用中间代码也有助于提高编译程序的可移植性。常用的中间代码有后缀式、三元式、四元式和树等形式。

参考答案

（49）D

试题（50）

以下关于程序错误的叙述中，正确的是__（50）__。

（50）A．编译正确的程序必然不包含语法错误

B．编译正确的程序必然不包含语义错误

C．除数为 0 的错误可以在语义分析阶段检查出来

D．除数为 0 的错误可以在语法分析阶段检查出来

试题（50）分析

本题考查程序语言基础知识。

编译程序的工作过程可以分为词法分析、语法分析、语义分析、中间代码生成、代码优化和目标代码生成等阶段。

用户编写的源程序不可避免地会有一些错误，这些错误大致可分为静态错误和动态错误。动态错误也称动态语义错误，它们发生在程序运行时，例如变量取零时作除数、引用数组元素下标错误等。静态错误是指编译阶段发现的程序错误，可分为语法错误和静态语义错误，如单词拼写错误、标点符号错、表达式中缺少操作数、括号不匹配等有关语言结构上的错误称为语法错误，而语义分析时发现的运算符与运算对象类型不合法等错误属于静态语义错误。

参考答案

（50）A

试题（51）、（52）

关系 R1 和 R2 如下图所示：

R1

A	B	C	D
a	d	c	e
c	b	a	e
d	e	c	e
e	f	d	a

R2

C	D	E	F
a	e	c	a
a	e	a	b
c	e	b	c

若进行 R1⋈ R2 运算，则结果集为__（51）__元关系，共有__（52）__个元组。

（51）A．4　　B．5　　C．6　　D．7

（52）A．4　　B．5　　C．6　　D．7

试题（51）、（52）分析

本题考查数据库系统中关系代数运算方面的基础知识。

试题（51）的正确选项为 C。根据题意 $R1 \bowtie R2$ 为自然联接，自然联接是一种特殊的等值联接，它要求两个关系中进行比较的分量必须是相同的属性，并且在结果集中将重复属性列去掉，R2.C 和 R2.D 去掉，故结果集为 6 元关系。

试题（52）的正确选项为 A。本题比较的条件为“R1.C=R2.C∧R1.D=R2.D”，从下图所示的 R1×R2 的结果集中可见，共有 4 个元组满足条件，分别是第 3 个、第 4 个、第 5 个和第 9 个元组。

R1×R2

R1.A	R1.B	R1.C	R1.D	R2.C	R2.D	R2.E	R2.F
a	d	c	e	a	e	c	a
a	d	c	e	a	e	a	b
a	d	c	e	c	e	b	c
c	b	a	e	a	e	c	a
c	b	a	e	a	e	a	b
c	b	a	e	c	e	b	c
d	e	c	e	a	e	c	a
d	e	c	e	a	e	a	b
d	e	c	e	c	e	b	c
e	f	d	a	a	e	c	a
e	f	d	a	a	e	a	b
e	f	d	a	c	e	b	c

参考答案

（51）C　（52）A

试题（53）、（54）

设有关系模式 R（E，N，M，L，Q），其函数依赖集为 F={ E→N，EM→Q，M→L}。则关系模式 R 达到了__（53）__；该关系模式__（54）__。

（53）A．1NF　B．2NF　C．3NF　D．BCNF

（54）A．无须进行分解，因为已经达到了 3NF

B．无须进行分解，因为已经达到了 BCNF

C．尽管不存在部分函数依赖，但还存在传递依赖，所以需要进行分解

D．需要进行分解，因为存在冗余、修改操作的不一致性、插入和删除异常

试题（53）、（54）分析

本题考查关系数据库方面的基础知识。

试题（53）的正确选项为 A。根据题意 R 关系中的 EM 可决定该关系的所有属性，所以 R 关系的主键为 EM；又因为，EM→Q，而 E→N，M→L，可以得出 N 和 L 都部分依赖于码，所以，该关系模式属于 1NF。

试题（54）的正确选项为 D。关系模式 R 属于 1NF，1NF 存在冗余度大、修改操作的不一致性、插入异常和删除异常四个问题，所以 R 需要进行分解。

参考答案

（53）A　（54）D

试题（55）、（56）

已知关系模式：图书（图书编号，图书类型，图书名称，作者，出版社，出版日期，ISBN），图书编号唯一识别一本图书。建立“计算机”类图书的视图 Computer-BOOK，并要求进行修改、插入操作时保证该视图只有计算机类的图书。实现上述要求的 SQL 语

句如下：

```
CREATE ___(55)___
    AS SELECT 图书编号，图书名称，作者，出版社，出版日期
        FROM 图书
        WHERE 图书类型='计算机'
        ______(56)______;
```

（55）A. TABLE Computer-BOOK　　B. VIEW Computer-BOOK
　　　C. Computer-BOOK TABLE　　D. Computer-BOOK VIEW

（56）A. FOR ALL　　B. PUBLIC
　　　C. WITH CHECK OPTION　　D. WITH GRANT OPTION

试题（55）、（56）分析

本题考查数据库系统中关系代数运算方面的基础知识。

试题（55）的正确选项为 B，试题（56）的正确选项为 C。因为创建视图的语句格式如下：

```
CREATE VIEW 视图名 (列表名)
            AS SELECT 查询子句
            [WITH CHECK OPTION];
```

其中，WITH CHECK OPTION 表示对 UPDATE,INSTER,DELETE 操作时保证更新、插入或删除的行满足视图定义中的谓词条件（即子查询中的条件表达式）。另外，组成视图的属性列名或者全部省略或者全部指定。如果省略属性列名，则隐含该视图由 SELECT 子查询目标列的主属性组成。

可见，完整的 Computer-BOOK 视图创建语句如下：

```
CREATE VIEW Computer-BOOK
    AS SELECT 图书编号，图书名称，作者，出版社，出版日期
    FROM 图书
    WHERE 图书类型='计算机'
    WITH CHECK OPTION;
```

参考答案

（55）B　　（56）C

试题（57）

在字符串的模式匹配过程中，如果模式串的每个字符依次和主串中一个连续的字符序列相等，则称为匹配成功。如果不能在主串中找到与模式串相同的子串，则称为匹配失败。在布鲁特—福斯模式匹配算法（朴素的或基本的模式匹配）中，若主串和模式串的长度分别为 n 和 m（且 n 远大于 m），且恰好在主串末尾的 m 个字符处匹配成功，则

在上述的模式匹配过程中，字符的比较次数最多为__（57）__。

（57）A. n*m　　B. (n-m+1)*m　　C. (n-m-1)*m　　D. (n-m)*n

试题（57）分析

本题考查数据结构基础知识。

假设主串和模式串的长度分别为 n 和 m，位置序号从 0 开始计算。设从主串的第 i 个位置开始与模式串匹配成功，在前 i 趟匹配中（位置 0～i–1），每趟不成功的匹配都是模式串的第一个字符与主串中相应的字符不相同，则在前 i 趟匹配中，字符的比较共进行了 i 次，而第 i+1 趟（从位置 i 开始）成功匹配的字符比较次数为 m，所以总的字符比较次数为 i+m（0≤i≤n–m）。

而在最坏情况下，每一趟不成功的匹配都是模式串的最后一个字符与主串中相应的字符不相等，则主串中新一趟的起始位置为 i–m+2。若设从主串的第 i 个字符开始匹配时成功，则前 i 趟不成功的匹配中，每趟都比较了 m 次，总共比较了 i×m 次，第 i+1 趟的成功匹配也比较了 m 次。因此，最坏情况下的比较次数为(n–m+1)*m。

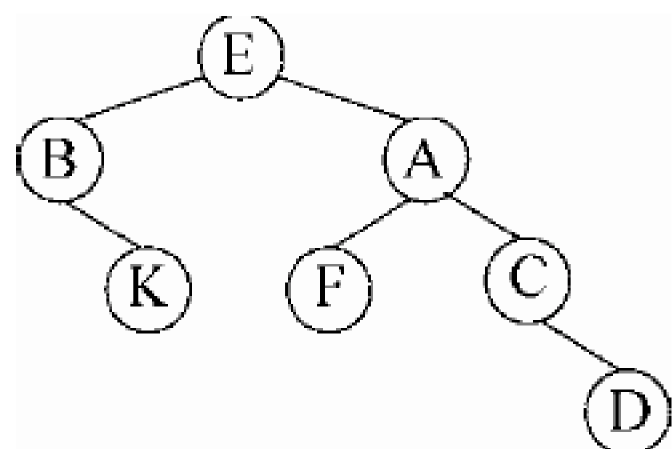
参考答案

（57）B

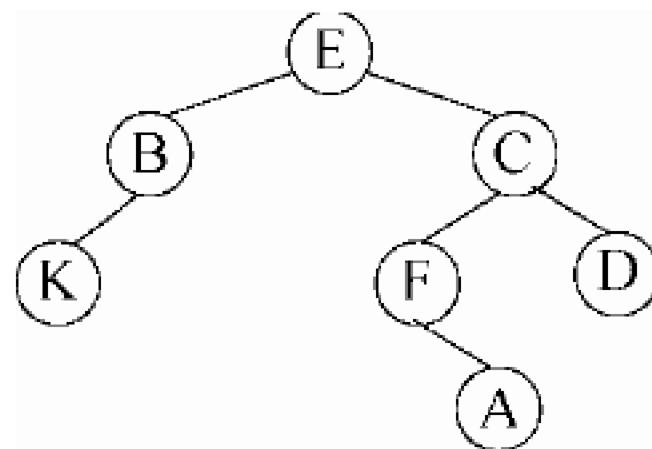
试题（58）

若某二叉树的后序遍历序列为 KBFDCAE，中序遍历序列为 BKEFACD，则该二叉树为__（58）__。

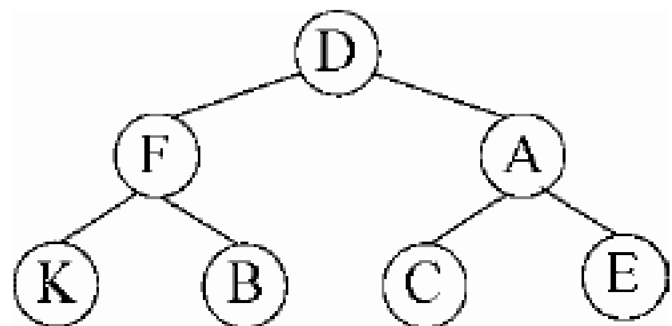
（58）A.



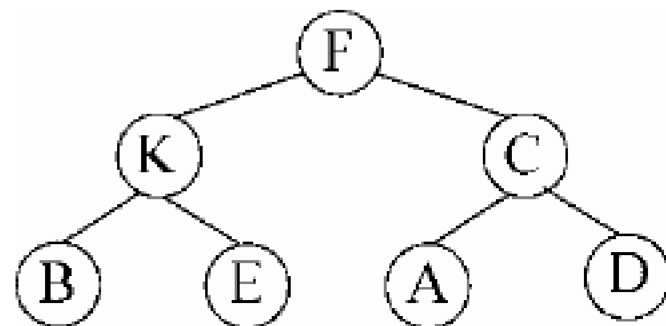
B.



C.



D.



试题（58）分析

本题考查数据结构基础知识。

根据后序遍历序列 KBFDCAE，可以确定根结点为 E，然后根据中序遍历序列（为 BKEFACD），可以确定 B、K 为左子树的结点，F、A、C、D 是右子树的结点。再根据左子树的后序遍历序列为 KB、中序遍历序列为 BK，所以可以确定 B 是左子树的根结点，K 是结点 B 的右子树上。同理可推出其他结点的位置。

参考答案

（58）A

试题（59）

在 13 个元素构成的有序表 M[1..13]中进行折半查找（向下取整），若找到的元素为 M[4]，则被比较的元素依次为__（59）__。

（59）A. M[7]、M[3]、M[5]、M[4]　　B. M[7]、M[5]、M[4]

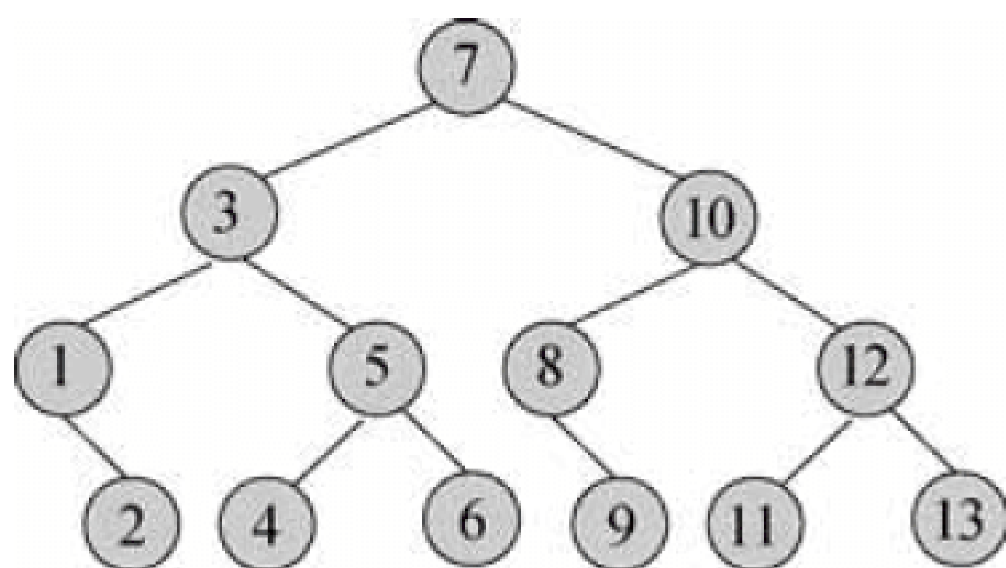
C. M[7]、M[6]、M[4]　　D. M[7]、M[4]

试题（59）分析

本题考查数据结构基础知识。

设查找表的元素存储在一维数组 r[1..n]中，在表中的元素已经按关键字递增方式排序的情况下，进行折半查找的方法是：首先将待查元素的关键字（key）值与表 r 中间位置上（下标为 mid）记录的关键字进行比较，若相等，则查找成功；若 key>r[mid].key，则说明待查记录只可能在后半个子表 r[mid+1..n]中，下一步应在后半个子表中进行查找，若 key<r[mid].key，说明待查记录只可能在前半个子表 r[1..mid－1]中，下一步应在 r 的前半个子表中进行查找，通过逐步缩小范围，直到查找成功或子表为空时失败为止。

折半查找的过程可以用一棵二叉树描述，方法是以当前查找区间的中间位置序号作为根，左半个子表和右半个子表中的记录序号分别作为根的左子树和右子树上的节点，具有 13 个节点的折半查找判定树如下图所示。

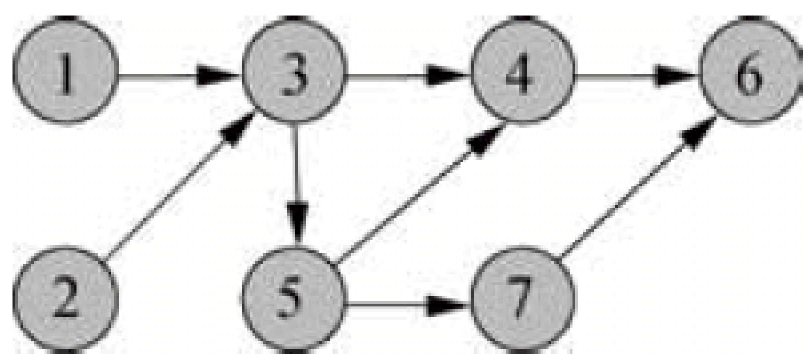


参考答案

（59）A

试题（60）

拓扑排序是将有向图中所有顶点排成一个线性序列的过程，并且该序列满足：若在 AOV 网中从顶点 v_i 到 v_j 有一条路径，则顶点 v_i 必然在顶点 v_j 之前。对于下面所示的有向图，__（60）__是其拓扑序列。



（60）A. 1 2 3 4 5 7 6　　B. 1 2 3 5 4 6 7

C. 2 1 3 5 4 7 6　　D. 2 1 3 4 5 6 7

试题（60）分析

本题考查数据结构基础知识。

拓扑排序是将 AOV 网中所有顶点排成一个线性序列的过程，并且该序列满足：若在 AOV 网中从顶点 v_i 到 v_j 有一条路径，则在该线性序列中，顶点 v_i 必然在顶点 v_j 之前。

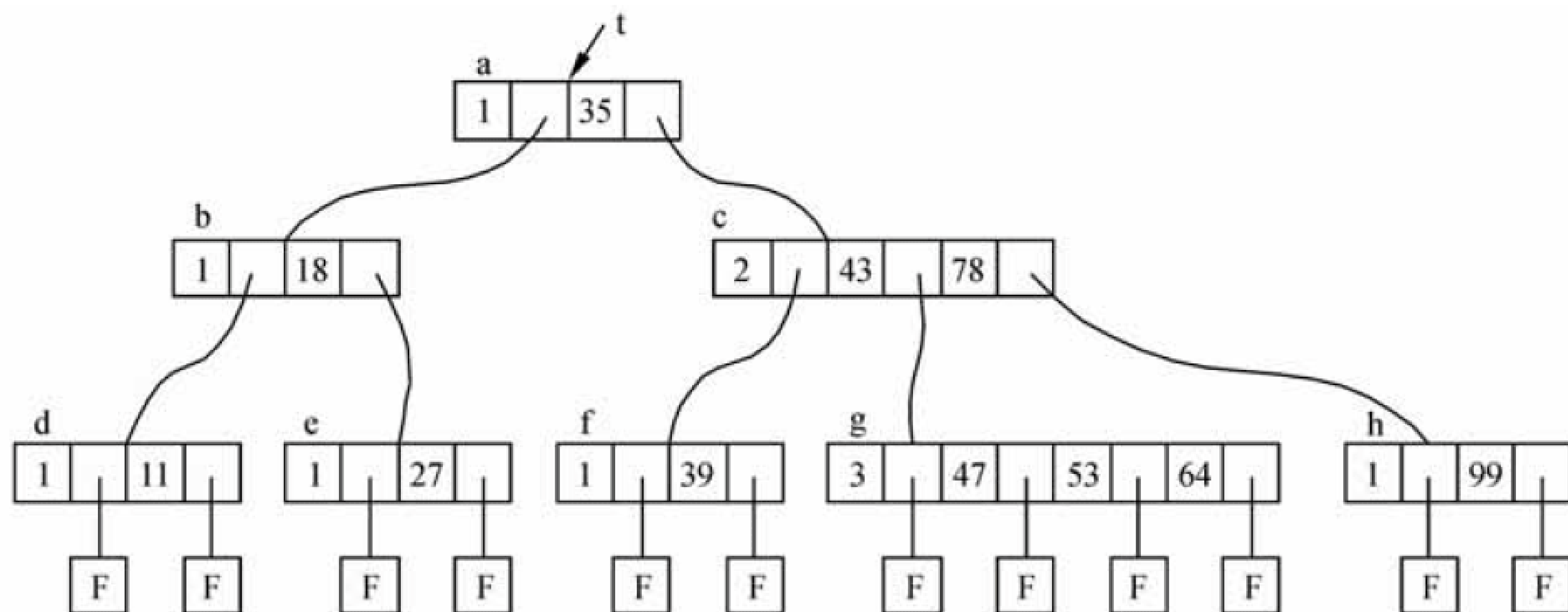
题中所示有向图的拓扑序列有：1235476，2135476，1235746，2135746。

参考答案

（60）C

试题（61）

下图所示为一棵 M 阶 B-树，M 最有可能的值为__（61）__。



（61）A．1　　B．2　　C．3　　D．4

试题（61）分析

本题考查数据结构基础知识。

在 m 阶 B-树的定义中，要求：

① 树中每个节点至多有 m 棵子树。

② 若根节点不是叶子节点，则至少有两棵子树。

③ 除根之外的所有非终端节点至少有 $\left\lceil \frac{m}{2} \right\rceil$ 棵子树。

因此，题图中所示 B-树最可能为 4 阶 B 树。

参考答案

（61）D

试题（62）、（63）

将数组{1, 1, 2, 4, 7, 5}从小到大排序，若采用__（62）__排序算法，则元素之间需要进行的比较次数最少，共需要进行__（63）__次元素之间的比较。

（62）A．直接插入　　B．归并　　C．堆　　D．快速

（63）A．5　　B．6　　C．7　　D．8

试题（62）、（63）分析

输入数组{1,1,2,4,7,5}基本有序（从小到大），在这种情况下，插入排序算法的时间

复杂度为 O(n)，归并排序和堆排序的时间复杂度为 O(nlgn)，而快速排序的时间复杂度为 $O(n^2)$。用插入排序算法排序该输入数组，第二个元素 1 需要和第一个元素 1 进行 1 次比较，第三个元素 2 需要和第二个元素 1 进行 1 次比较，第四个元素 4 需要和第三个元素 2 进行 1 次比较，第五个元素 7 需要和第四个元素 4 进行 1 次比较，第六个元素 5 需要和第五个元素 7 进行 1 次比较，比 7 小，和元素 7 交换，在和第四个元素 4 进行 1 次比较，得到最终的排序结果。因此一共需要进行 6 次比较。

参考答案

（62）A　（63）B

试题（64）、（65）

霍夫曼编码将频繁出现的字符采用短编码，出现频率较低的字符采用长编码。具体的操作过程为：i)以每个字符的出现频率作为关键字构建最小优先级队列；ii) 取出关键字最小的两个结点生成子树，根节点的关键字为孩子节点关键字之和，并将根节点插入到最小优先级队列中，直至得到一颗最优编码树。

霍夫曼编码方案是基于＿（64）＿策略的。用该方案对包含 a 到 f 六个字符的文件进行编码，文件包含 100,000 个字符，每个字符的出现频率（用百分比表示）如下表所示，则与固定长度编码相比，该编码方案节省了＿（65）＿存储空间。

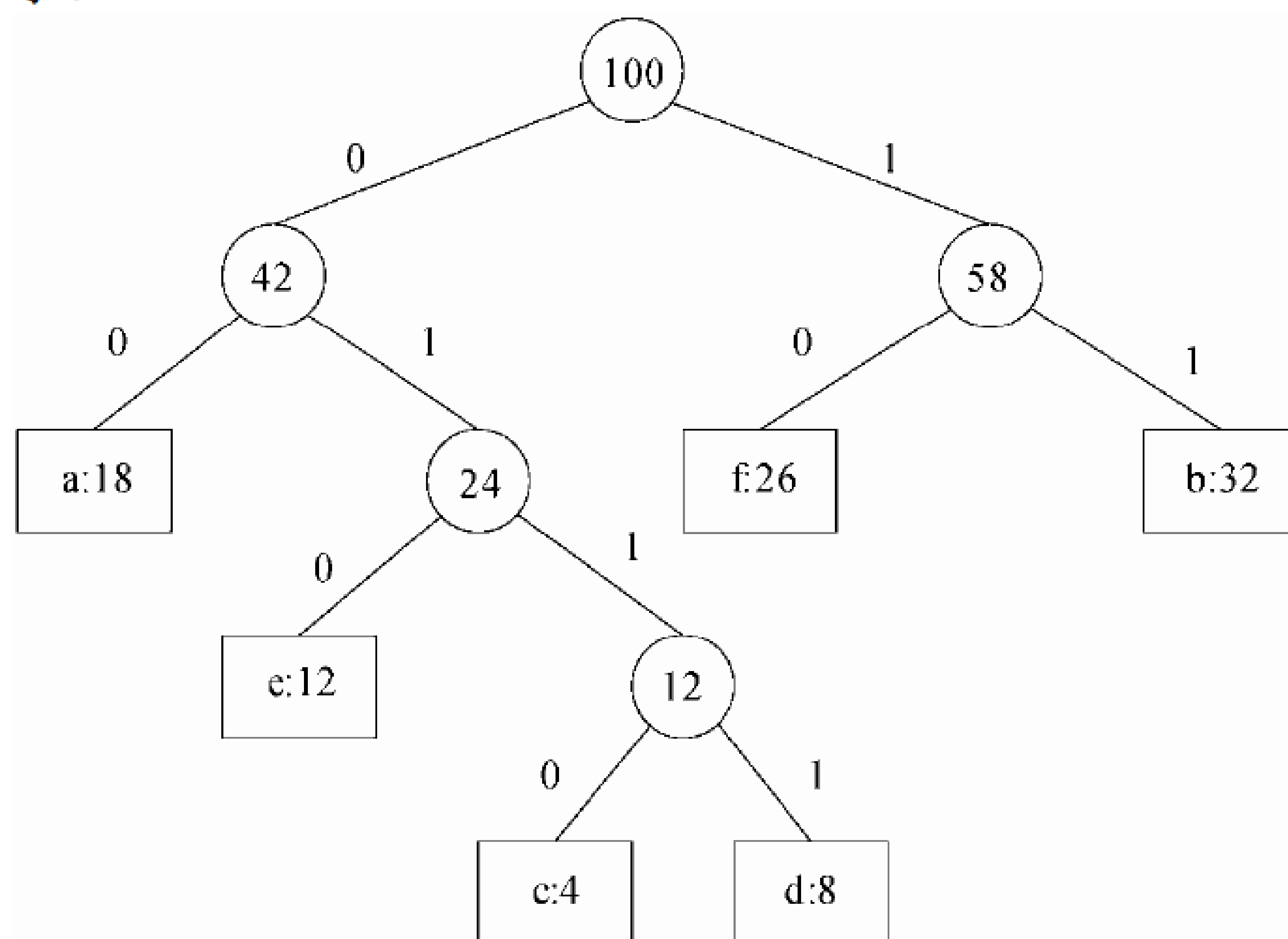
字符	a	b	c	d	e	f
出现频率	18	32	4	8	12	26

（64）A．分治　B．贪心　C．动态规划　D．回溯

（65）A．21%　B．27%　C．18%　D．36%

试题（64）、（65）分析

霍夫曼编码在构造最优编码树时，每次总是取出权值最小的两个结点来生成它们的父结点（新的中间结点），因此霍夫曼编码方案是基于贪心策略的。对应该实例构造最优编码树如下图所示。



实例中包含 6 个字符，若用定长编码，则需要三位，对包含 100,000 个字符的文件，需要 3*100,000=300,000 位的存储空间。而采用霍夫曼编码，则需要$(18+26+32)*2+12*3+(4+8)*4=236,000$ 位的存储空间，节省了 21%的存储空间。

参考答案

（64）B （65）A

试题（66）

在 Windows 2003 Server 中启用配置 SNMP 服务时，必须以__（66）__身份登录才能完成 SNMP 服务的配置功能。

（66）A．guest B．普通用户
C．administrator 组成员 D．user 组成员

试题（66）分析

本题考查 Windows 2003 中有关 SNMP 服务配置的操作权限。

Windows Server 2003 中配置 SNMP 服务时，必须以管理员身份或者 Administrators 组成员身份登录才能完成 SNMP 服务的配置功能。一般用户或者普通用户不能完成 SNMP 配置服务。

参考答案

（66）C

试题（67）

在 ASP 的内置对象中，__（67）__对象可以修改 cookie 中的值。

（67）A．request B．response C．application D．session

试题（67）分析

本题考查 ASP 的内置对象的基础知识。在 ASP 的内置对象中，response 对象和 request 对象和 Cookie 有关。其中，request 对象中的 Cookies 集合是服务器根据用户的请求，发出的所有 Cookie 的值的集合，这些 Cookie 仅对相应的域有效，每个成员均为只读。Response 对象中的 Cookies 集合是服务器发回客户端的所有 Cookie 的值，这个集合为只写，所以只有 Response 对象可以修改 Cookie 中的值。

参考答案

（67）B

试题（68）、（69）

分配给某公司网络的地址块是 220.17.192.0/20，该网络被划分为__（68）__个 C 类子网，不属于该公司网络的子网地址是__（69）__。

（68）A．4 B．8 C．16 D．32

（69）A．220.17. 203. 0 B．220.17. 205. 0
C．220.17. 207. 0 D．220.17. 213. 0

试题（68）、（69）分析

220.17.192.0 是一个 C 类网络地址，应该有 24 位子网掩码，现在仅采用 20 位子网掩码，少了 4 位，所以被划分成了 16 个子网。

这 16 个子网号的第三个字节都应该在 192+0～192+15 之间，由于 213 大于 192+15，所以 220.17. 213. 0 不属于地址块 220.17.192.0/20。

参考答案

（68）C　　（69）D

试题（70）

如果 DNS 服务器更新了某域名的 IP 地址，造成客户端域名解析故障，在客户端可以用两种方法解决此问题，其中一种是在 Windows 命令行下执行__（70）__命令。

（70）A．ipconfig /all　　　B．ipconfig /renew

　　　C．ipconfig/flushdns　　　D．ipconfig/release

试题（70）分析

本题考查 Windows 系统中 DNS 服务及相关配置命令的基础操作。

ipconfig 命令详解如下：

① 具体功能

该命令用于显示所有当前的 TCP/IP 网络配置值、刷新动态主机配置协议（DHCP）和域名系统（DNS）设置。使用不带参数的 IPCONFIG 可以显示所有适配器的 IP 地址、子网掩码、默认网关。

② 语法详解

ipconfig [/all][/renew [adapter] [/release [adapter] [/flushdns] [/displaydns] [/registerdns] [/showclassidpadapter] [/setclassidpadapter][classID]

③ 参数说明

/all 显示所有适配器的完整 TCP/IP 配置信息。在没有该参数的情况下 IPCONFIG 只显示 IP 地址、子网掩码和各个适配器的默认网关值。适配器可以代表物理接口（例如安装的网络适配器）或逻辑接口（例如拨号连接）。

/renew 更新所有适配器（如果未指定适配器），或特定适配器（如果包含了 adapter 参数）的 DHCP 配置。该参数仅在具有配置为自动获取 IP 地址的网卡的计算机上可用。要指定适配器名称，请输入使用不带参数的 IPCONFIG 命令显示的适配器名称。

/release[adapter]发送 DHCPRELEASE 消息到 DHCP 服务器，以释放所有适配器（如果未指定适配器）或特定适配器（如果包含了 adapter 参数）的当前 DHCP 配置并丢弃 IP 地址配置。该参数可以禁用配置为自动获取 IP 地址的适配器的 TCP/IP。要指定适配器名称，请输入使用不带参数的 IPCONFIG 命令显示的适配器名称。

/flushdns 清理并重设 DNS 客户解析器缓存的内容。如有必要，在 DNS 疑难解答期间，可以使用本过程从缓存中丢弃否定性缓存记录和任何其他动态添加的记录。

DNS Client 服务为计算机解析和缓存 DNS 名称。为了要达到用最快速、最有效率

的方式，让客户端能够迅速找到网域的验证服务，在 Win2000/XP 系统中，加入了 DNS 快取（Cache）的功能。当第一次在找到了目的主机的 IP 地址后，操作系统就会将所查询到的名称及 IP 地址记录在本机的 DNS 快取缓冲区中，下次客户端还需要再查询时，就不需要到 DNS 服务器上查询，而直接使用本机 DNS Cache 中的数据即可，所以你查询的结果始终是同一 IP 地址。这个服务关闭后，dns 还可以解析，但是本地无法储存 dns 缓存。

参考答案

（70）C

试题（71）～（75）

Computers will become more advanced and they will also become easier to use. Improved speed recognition will make the operation of a computer easier. Virtual reality (虚拟现实), the technology of （71） with a computer using all of the human senses, will also contribute to better human and computer （72）. Other, exotic (奇异的) models of computation are being developed, including biological computing that uses living organisms, molecular computing that uses molecules with particular （73）, and computing that uses DNA, the basic unit of heredity (遗传), to store data and carry out operations. These are examples of possible future computational platforms that, so far, are limited in abilities or are strictly （74）. Scientists investigate them because of the physical limitations of miniaturizing circuits embedded in silicon. There are also （75） related to heat generated by even the tiniest of transistors.

（71）A. interact B. interacting C. communicate D. using
（72）A. interfaces B. behavior C. similarities D. comparison
（73）A. software B. properties C. programs D. hardware
（74）A. empirical B. real C. practical D. theoretical
（75）A. developments B. advantages C. limitations D. improvements

试题（71）～（75）分析

本题考查对英语资料的阅读理解。

计算机将会变得越来越高级而且更易于使用。识别速度的提升将使计算机的操作更加容易。虚拟现实是使用人的感觉与计算机进行交互的一种技术，它也会使得人机界面更好。另外，各种新奇的计算模型正在不断发展，如生物计算使用人的器官、分子计算使用具有特定属性的分子、DNA 计算采用遗传的基本单元来存储数据和执行操作。到目前为止，这些都还属于未来计算平台，能力非常有限，并且只限于理论方面的研究。科学家们之所以研究这些计算模型，原因在于嵌入硅片中的微型芯片的体积的局限性，以及即使是非常小的晶体管所产生的热量方面的局限性。

参考答案

（71）B （72）A （73）B （74）D （75）C

第 16 章　2012 下半年软件设计师下午试题分析与解答

试题一（共 15 分）

阅读下列说明和图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

【说明】

某电子商务系统采用以数据库为中心的集成方式改进购物车的功能，详细需求如下：

1．加入购物车。顾客浏览商品，点击加入购物车，根据商品标识从商品表中读取商品信息，并更新购物车表。

2．浏览购物车。顾客提交浏览购物车请求后，显示出购物车表中的商品信息。

3．提交订单。顾客点击提交订单请求，后台计算购物车表中商品的总价（包括运费）加入订单表，将购物车表中的商品状态改为待付款，显示订单详情。若商家改变价格，则刷新后可看到更改后的价格。

4．改变价格。商家查看订购自家商品的订单信息，根据特殊优惠条件修改价格，更新订单表中的商品价格。

5．付款。顾客点击付款后，系统先根据顾客表中关联的支付账户，将转账请求（验证码、价格等）提交给支付系统（如信用卡系统）进行转账；然后根据转账结果返回支付状态并更改购物车表中商品的状态。

6．物流跟踪。商家发货后，需按订单标识添加物流标识（物流公司、运单号）；然后可根据顾客或商家的标识以及订单标识，查询订单表中的物流标识，并从相应物流系统查询物流信息。

7．生成报表。根据管理员和商家设置的报表选项，从订单表、商品表以及商品分类表中读取数据，调用第三方服务 Crystal Reports 生成相关报表。

8．维护信息。管理员维护（增、删、改、查）顾客表、商品分类表和商品表中的信息。

现采用结构化方法实现上述需求，在系统分析阶段得到如图 1-1 所示的顶层数据流图和图 1-2 所示的 0 层数据流图。

【问题 1】（4 分）

使用说明中的词语，给出图 1-1 中的实体 E1～E4 的名称。

【问题 2】（4 分）

使用说明中的词语，给出图 1-2 中的数据存储 D1～D4 的名称。

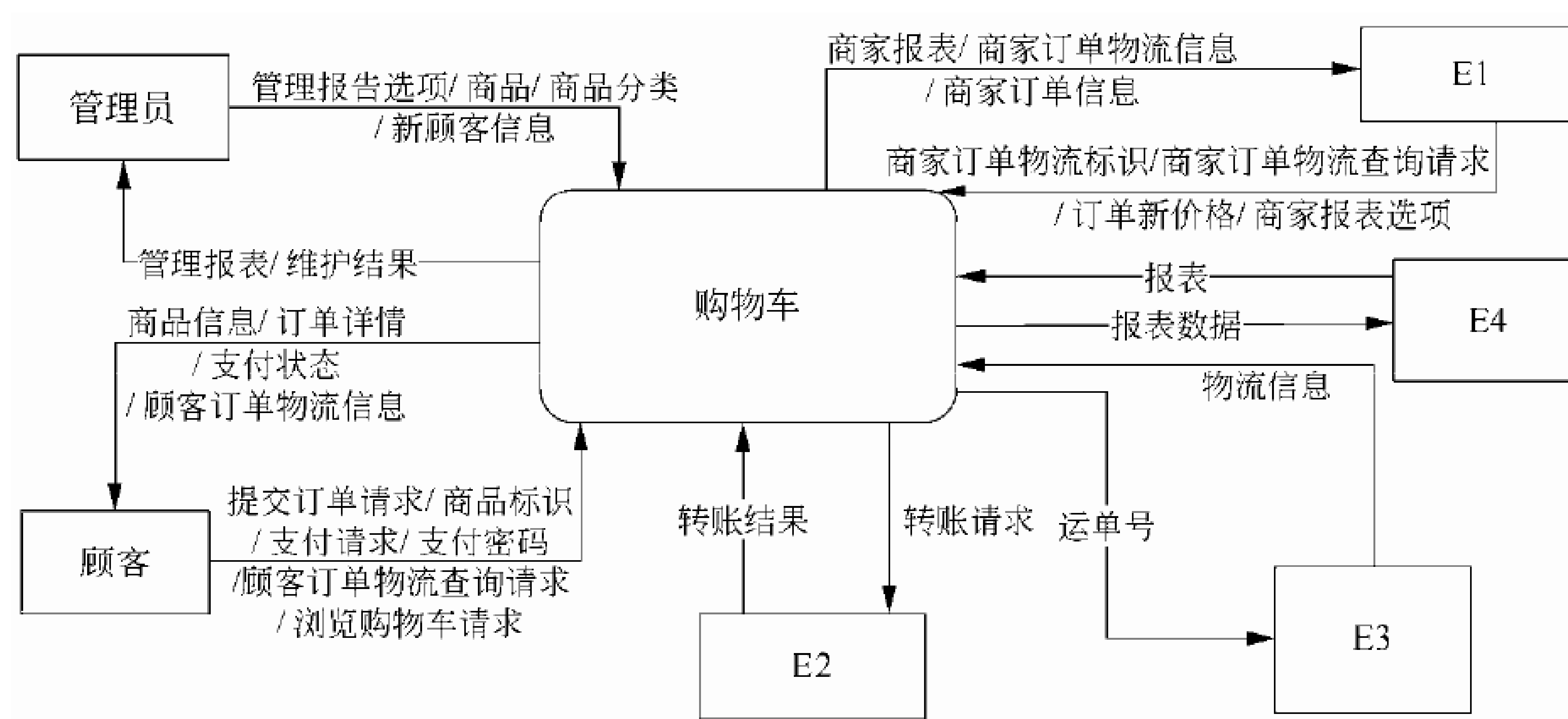


图 1-1　顶层数据流图

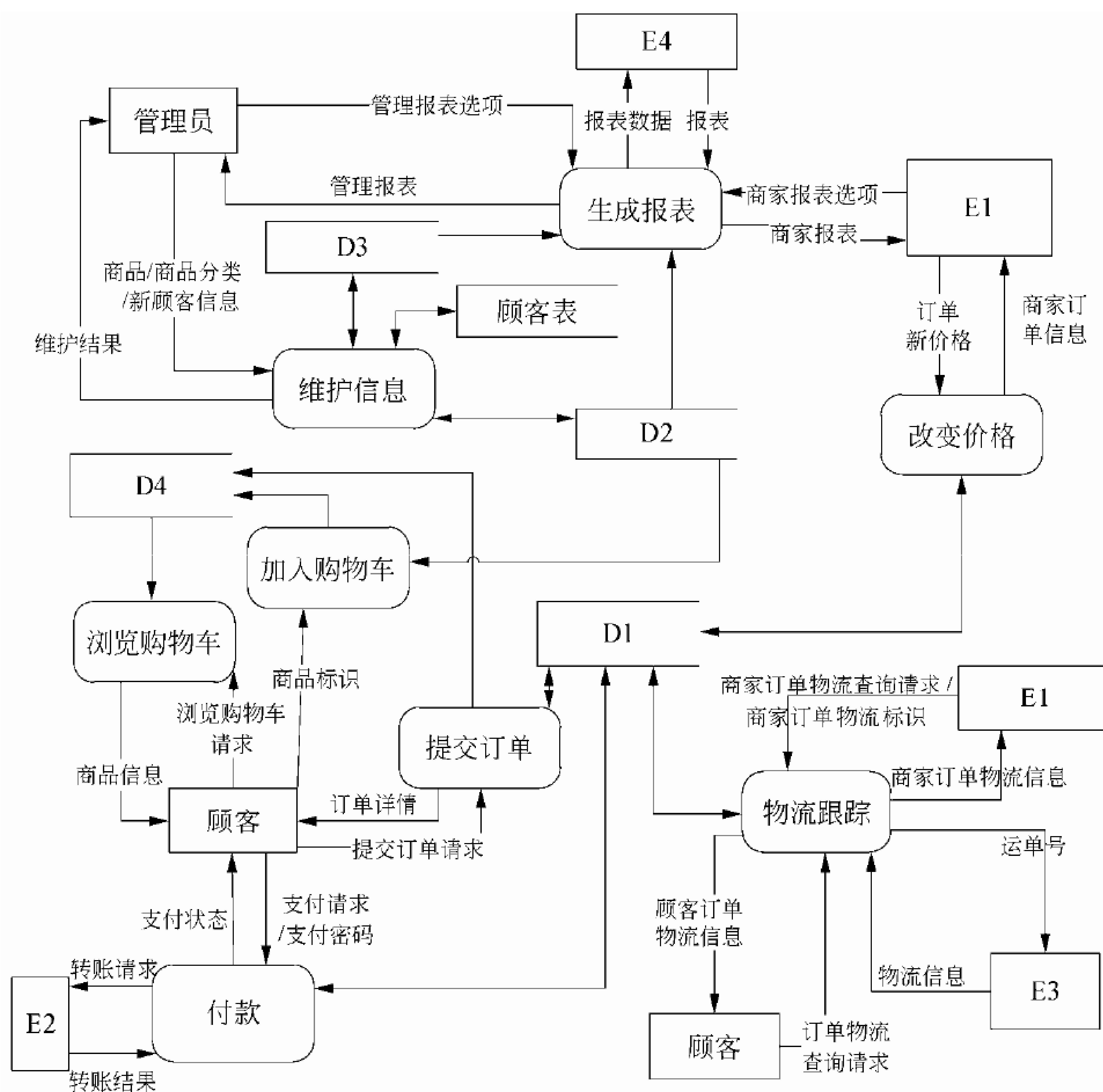


图 1-2　0 层数据流图

【问题 3】（4 分）

图 1-2 中缺失了数据流，请用说明或图 1-2 中的词语，给出其起点和终点。

【问题 4】（3 分）

根据说明，给出数据流“转账请求”“顾客订单物流查询请求”和“商家订单物流查询请求”的各组成数据项。

试题一分析

本题考查采用结构化方法进行系统分析与设计，主要考查数据流图（DFD）的应用，是比较传统的题目，要求考生细心分析题目中所描述的内容。

DFD 是一种便于用户理解、分析系统数据流程的图形化建模工具，是系统逻辑模型的重要组成部分。

【问题 1】

本问题考查顶层 DFD。顶层 DFD 一般用来确定系统边界，将待开发系统看作一个加工，图中只有唯一的一个处理和一些外部实体，以及这两者之间的输入输出数据流。题目要求根据描述确定图中的外部实体。外部实体可以是和系统交互的人，以及和系统交互的外部系统或服务。分析题目中的描述，并结合已经在顶层数据流图中给出的数据流进行分析。分析题目中的说明，管理员维护系统中信息，顾客和商家是系统的主要使用者；商家查看订购自家商品的订单信息，根据特殊优惠条件修改价格，更新订单表中的商品价格，还可以添加物流标识并进行物流跟踪；使用支付系统进行支付，通过物流系统进行物流跟踪，以及第三方服务 Crystal Report 生成报表。可以看出，和系统的交互者包括管理员、顾客、商家三类人，支付系统、物流系统和 Crystal Report 三种外部系统。

对应图 1-1 中数据流和实体的对应关系，管理员和顾客已经给出，可知 E1 为商家，E2 为支付系统，E3 为物流系统，E4 为第三方服务 Crystal Report。

【问题 2】

本问题考查 0 层 DFD 中数据存储的确定。根据说明中所描述的处理和相关数据存储之间的连接关系，判定每个数据存储。加入购物车和浏览购物车分别读取和更新购物车表中的数据；改变价格和提交订单要读取和更新订单表中的数据；维护信息时需要维护商品表和商品分类表，生成报告要读取商品表和商品分类表，加入购物车时，需要读取商品表中的商品信息。

根据描述和图 1-2 中的数据存储的输入输出数据流提示，可知：D1 为订单表，D2 为商品表，D3 为商品分类表，D4 为购物车表。

【问题 3】

本问题考查绘制 0 层 DFD 时是否将本层该绘制的数据流全部绘制出。对照顶层数据流图和 0 层数据流图，检查是否和外部实体之间的数据流一致；仔细对照说明中的描述和图 1-2 中给出的数据流，检查是否遗漏掉信息。说明中：提交订单处理时，后台计算购物车表中的商品的总价，即需要读出购物车表中的相关价格进行计算，读取出其中

数据；付款需要读取顾客表中关联的支付账户，并向支付系统提交转账请求，然后根据转账结果更改购物车表中商品的状态；生成报告时根据管理员和商家设置的报告选项，从订单表、商品表以及商品分类表中读取数据，再调用第三方服务 Crystal Reports 生成相关报告。将这些说明和图 1-2 进行对照，发现缺少了从付款到购物车表（D4）、从购物车表到提交订单、从顾客表到付款，以及从订单表（D1）到生成报表等 4 条数据流。

【问题 4】

本问题考查在绘制数据流图时数据流的数据项组成。数据流图描述了系统的分解，但它并没有给出图中各成分的说明。通常采用数据字典为数据流图中的每个数据流、文件、处理，以及组成数据流或文件的数据项做出说明。对于数据流，通常列出该数据流的各组成数据项，并采用数据字典定义式中出现的符号进行表示，如“=”表示“被定义为”，“+”表示“与”“{……}”表示其中数据可以有多个等等。本试题说明中：付款时，需根据顾客表中关联的支付账户将转账请求（验证码、价格等）提交给支付系统；物流跟踪时，根据顾客和商家的标识以及订单标识进行查询，而且在改变价格时商家查看订购自家商品的订单信息，可知商家可以查询一批订单。可以看出，提交给支付系统的请求中包含支付账户、验证码与价格；顾客订单查询请求中有顾客标识、订单标识；商家订单查询请求中有商家标识、订单标识（一批订购自家商品的订单标识）。因此“转账请求=支付账户+验证码+价格”；“商家订单物流查询请求=物流标识+{订单标识}”；“顾客订单物流标识=物流标识+订单标识”。

参考答案

【问题 1】

E1：商家　　E2：支付系统　　E3：物流系统

E4：Crystal Reports 或第三方服务

【问题 2】

D1：订单表　　D2：商品表　　D3：商品分类表　　D4：购物车表

【问题 3】

图 1-2 中缺少的数据流：

起点	终点
付款	D4 或购物车表
D4 或购物车表	提交订单
顾客表	付款
D1 或订单表	生成报表

【问题 4】

转账请求 = 验证码+价格+账号信息

顾客订单物流查询请求 = 顾客标识+订单标识

商家订单物流查询请求 = 商家标识+{订单标识}

试题二（共 15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某会议策划公司为了方便客户，便于开展和管理各项业务活动，需要构建一个基于网络的会议预定系统。

【需求分析】

1．会议策划公司设有受理部、策划部和其他部门。部门信息包括部门号、部门名称、部门主管、电话和邮箱号。每个部门有多名员工处理部门的日常事务，每名员工只能在一个部门工作。每个部门有一名主管负责管理本部门的事务和人员。

2．员工信息包括员工号、姓名、部门号、职位、联系方式和工资；其中，职位包括主管、业务员、策划员等。业务员负责受理会议申请。若申请符合公司规定，则置受理标志并填写业务员的员工号。策划部主管为已受理的会议申请制定策划任务，包括策划内容、参与人数、要求完成时间等。一个已受理的会议申请对应一个策划任务，一个策划任务只对应一个已受理的会议申请，但一个策划任务可由多名策划员参与执行，且一名策划员可以参与多项策划任务。

3．客户信息包括客户号、单位名称、通信地址、所属省份、联系人、联系电话、银行账号。其中，一个客户号唯一标识一个客户。一个客户可以提交多个会议申请，但一个会议申请对应唯一的一个客户号。

4．会议申请信息包括申请号、开会日期、会议地点、持续天数、会议人数、预算费用、会议类型、酒店要求、会议室要求、客房类型、客房数、联系人、联系方式、受理标志和业务员的员工号等。客房类型有豪华套房、普通套房、标准间、三人间等，且申请号和客房类型决定客房数。

【概念模型设计】

根据需求阶段收集的信息，设计的实体联系图和关系模式（不完整）如下：

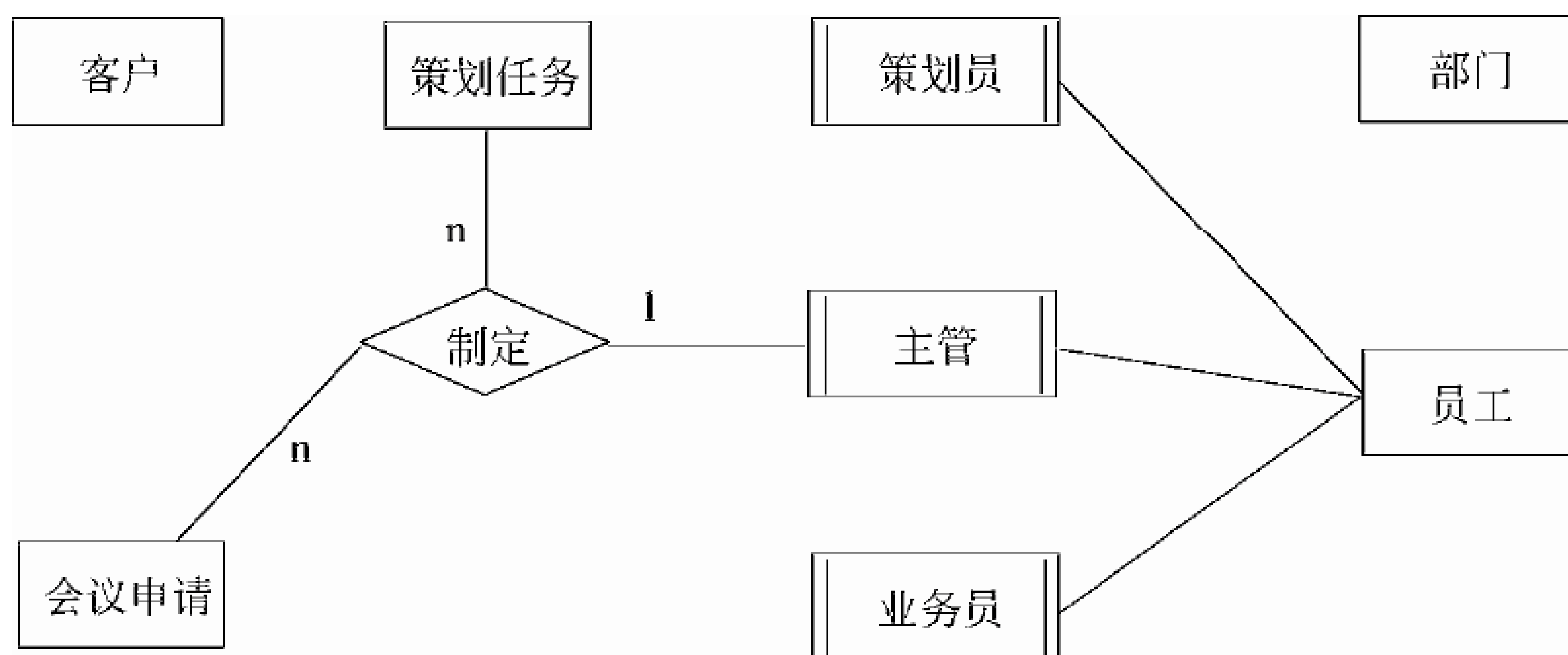


图 2-1　实体联系图

【关系模式设计】

部门（部门号，部门名称，主管，电话，邮箱号）
员工（员工号，姓名，____(a)____，联系方式，工资）
客户（客户号，单位名称，通信地址，所属省份，联系人，联系电话，银行账号）
会议申请（____(b)____，开会日期，会议地点，持续天数，会议人数，预算费用，会议类型，酒店要求，会议室要求，客房数，联系人，联系方式，受理标志，员工号）
策划任务（____(c)____，策划内容，参与人数，要求完成时间）
执行策划（____(d)____，实际完成时间）

【问题 1】（5 分）

根据问题描述，补充五个联系、联系的类型，完善图 2-1 的实体联系图。

【问题 2】（7 分）

根据实体联系图，将关系模式中的空（a）~（d）补充完整（1 个空缺处可能有多个数据项）。对会议申请、策划任务和执行策划关系模式，用下划线和#分别指出各关系模式的主键和外键。

【问题 3】（3 分）

请说明关系模式“会议申请”存在的问题及解决方案。

试题二分析

本题考查数据库设计方面的应用知识。

【问题 1】

根据题意，一个客户可以提交多个会议申请，但一个会议申请对应唯一的一个客户号，故应在客户和会议申请之间增加一个 1∶n 的“提交”联系；由于业务员负责受理会议申请，若申请符合公司规定则置受理标志并填写业务员的员工号，因此业务员和会议申请之间有一个 1∶n 的“受理”联系；由于一个已受理的会议申请对应一个策划任务，一个策划任务只对应一个已受理的会议申请，但一个策划任务可由多名策划员参与执行，且一名策划员可以参与多项策划任务，因此策划任务和策划员之间有一个 n∶m 的“执行”联系；由于每个部门有多名员工处理部门的日常事务，每名员工只能在一个部门工作，因此部门和员工之间有一个 1∶n 的“所属”联系；又由于每个部门有一名主管负责管理本部门的事务和人员，而该主管也是一名员工，因此主管和部门之间有一个 1∶1 的“管理”联系。

根据上述分析，完善图 2-1 所示的实体联系图可参见参考答案。

【问题 2】

根据题意，在员工关系模式中，部门与员工之间是一个 1∶n 的联系，需要将 1 端（即部门）的码“部门号”并入员工关系；又因为每个员工担任相应职位，故员工关系模式欢迎添加“职位”属性；可见，空（a）应填写“部门号，职位”。

在会议申请关系模式中，由于申请号、客房类型、客户号为主键，故空（b）应填写“申请号，客房类型，客户号”；在策划任务关系模式中，申请号、员工号为主键，故空（c）应填写“申请号，客户号”；由于一个策划任务可由多名策划员参与执行，且一名策划员可以参与多项策划任务，故在执行策划关系模式中，执行策划又由于一个业务员可

以安排多个托运申请，申请号、员工号为主键，故空（d）应填写“申请号，客户号”。

会议申请关系模式的主键为“申请号，客房类型”，因为，申请号、客房类型能唯一标识该关系模式的每一个元组。会议申请关系模式的外键为客户号及员工号，因为，客户号及员工号分别为客户及员工关系模式的主键，故为该关系模式的外键。

策划任务关系模式的主键为申请号，因为，申请号能唯一标识该关系模式的每一个元组，故申请号为该关系模式的主键。策划任务关系模式的外键为员工号，因为，员工号为员工关系模式的主键，故为该关系模式的外键。

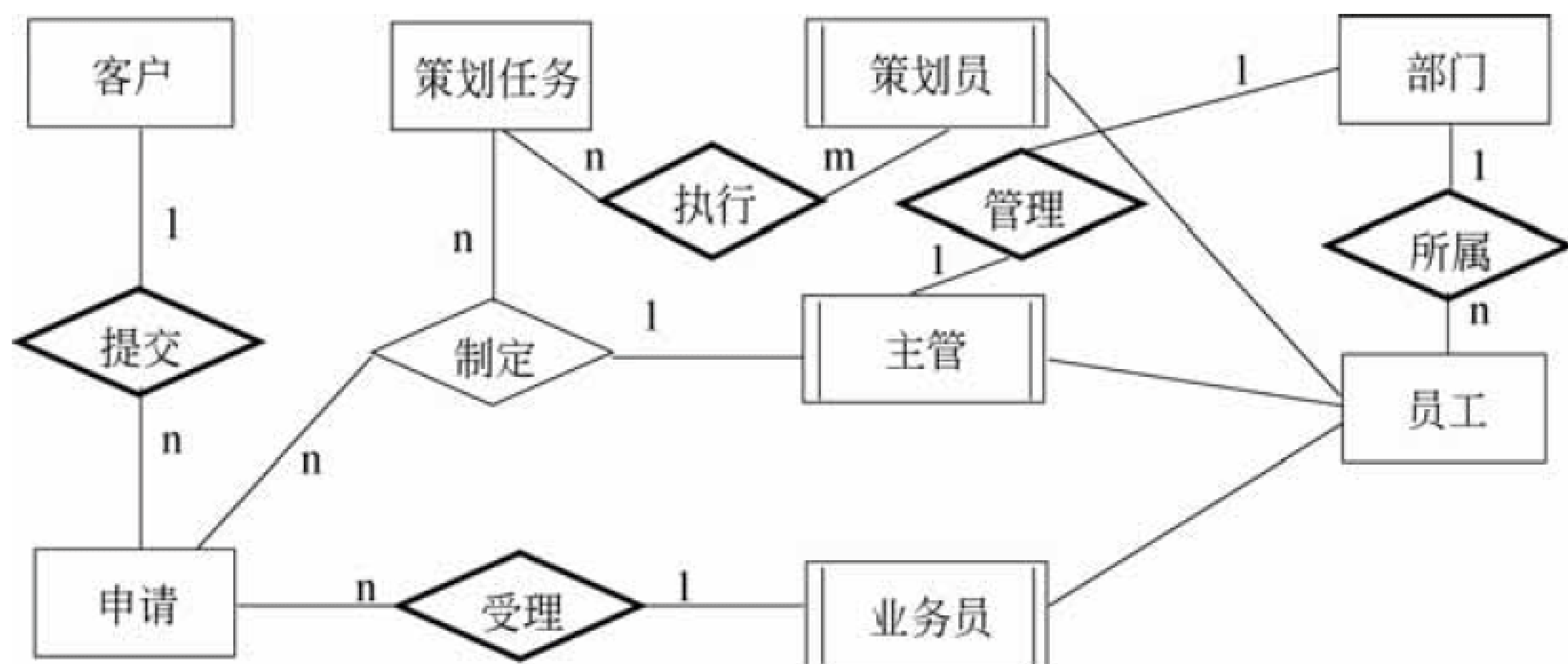
执行策划关系模式的主键为“申请号，员工号”，因为，申请号及员工号能唯一标识该关系模式的每一个元组，故“申请号，员工号”为该关系模式的主键。执行策划关系模式的外键为申请号及员工号，因为，申请号和员工号分别为会议申请和员工关系模式的主键，故为该关系模式的外键。

【问题 3】

关系模式“会议申请”存在数据冗余及数据修改的不一致性问题，应该将关系模式分解，分解后的关系模式参见参考答案。

参考答案

【问题 1】



【问题 2】

（a）部门号，职位

（b）申请号，客房类型，客户号

（c）申请号，员工号

（d）申请号，员工号

关系模式为：

会议申请（申请号，客房类型，客户号#，开会日期，会议地点，持续天数，会议人数，预算费用，会议类型，酒店要求，会议室要求，客房数，联系人，联系方式，受理标志，员工号#）

策划任务（申请号#，员工号#，策划内容，参与人数，要求完成时间）

执行策划（申请号#，员工号#，实际完成时间）

【问题 3】

会议申请存在数据冗余及数据修改的不一致性问题，应该将关系模式分解为如下两个模式：

会议申请 1（申请号，客户号，开会日期，会议地点，持续天数，会议人数，预算费用，会议类型，酒店要求，会议室要求，联系人，联系方式，受理标志，员工号）

会议申请 2（申请号，客房类型，客房数）。

试题三（15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某城市的各国家公园周边建造了许多供游客租用的小木屋和营地，为此，该城市设置了一个中心售票处和若干个区域售票处。游客若想租用小木屋或营地，必须前往中心售票处进行预定并用现金支付全额费用。所有的预定操作全部由售票处的工作人员手工完成。现欲开发一信息系统，实现小木屋和营地的预定及管理功能，以取代手工操作。该系统的主要功能描述如下：

1．管理预定申请。游客可以前往任何一个售票处提出预定申请。系统对来自各个售票处的预定申请进行统一管理。

2．预定。预定操作包含登记游客预定信息、计算租赁费用、付费等步骤。

3．支付管理。游客付费时可以选择现金和信用卡付款两种方式。使用信用卡支付可以享受 3%的折扣，现金支付没有折扣。

4．游客取消预定。预定成功之后，游客可以在任何时间取消预定，但需支付赔偿金，剩余部分则退还给游客。赔偿金的计算规则是，在预定入住时间之前的 48 小时内取消，支付租赁费用 10%的赔偿金；在预定入住时间之后取消，则支付租赁费用 50%的赔偿金。

5．自动取消预定。如果遇到恶劣天气（如暴雨、山洪等），系统会自动取消所有的预定，发布取消预定消息，全额退款。

6．信息查询。售票处工作人员查询小木屋和营地的预定情况和使用情况，以判断是否能够批准游客的预定申请。

现采用面向对象方法开发上述系统，得到如表 3-1 所示的用例列表和表 3-2 所示的类列表。对应的用例图和类图分别如图 3-1 和图 3-2 所示。

表 3-1 用例列表

用 例 名	说 明	用 例 名	说 明
ManageInquiries	管理预定申请	ManageCashPayment	现金支付
MakeReservation	预定	ManageCrCardPayment	信用卡支付
ManagePayment	支付管理	GetDiscount	计算付款折扣
CancelReservation	游客取消预定	AutoCancelReservation	系统自动取消预定
CheckAvailability	信息查询	CalculateRefund	计算取消预定的赔偿金
PublishMessage	发布取消预定消息		

表 3-2　类列表

类　　名	说　　明	类　　名	说　　明
NationalPark	国家公园	Customer	游客
Reservation	预定申请	ReservationItem	预定申请内容
TicketingOfficer	售票处	CampSite	营地
Bungalow	小木屋	Payment	付款
Discount	付款折扣	CashPayment	现金支付
CreditCardPayment	信用卡支付	Rate	租赁费用

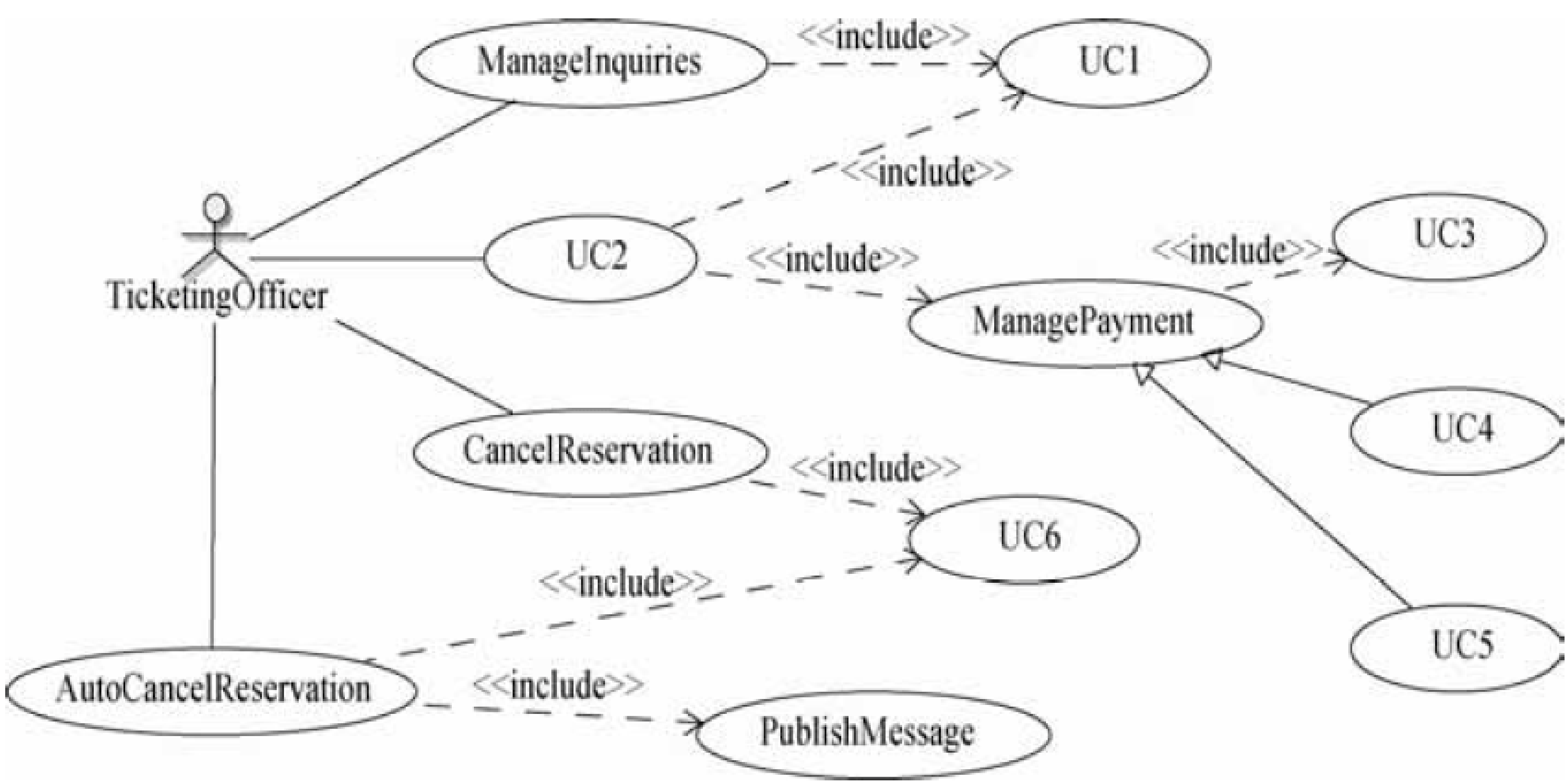


图 3-1　用例图

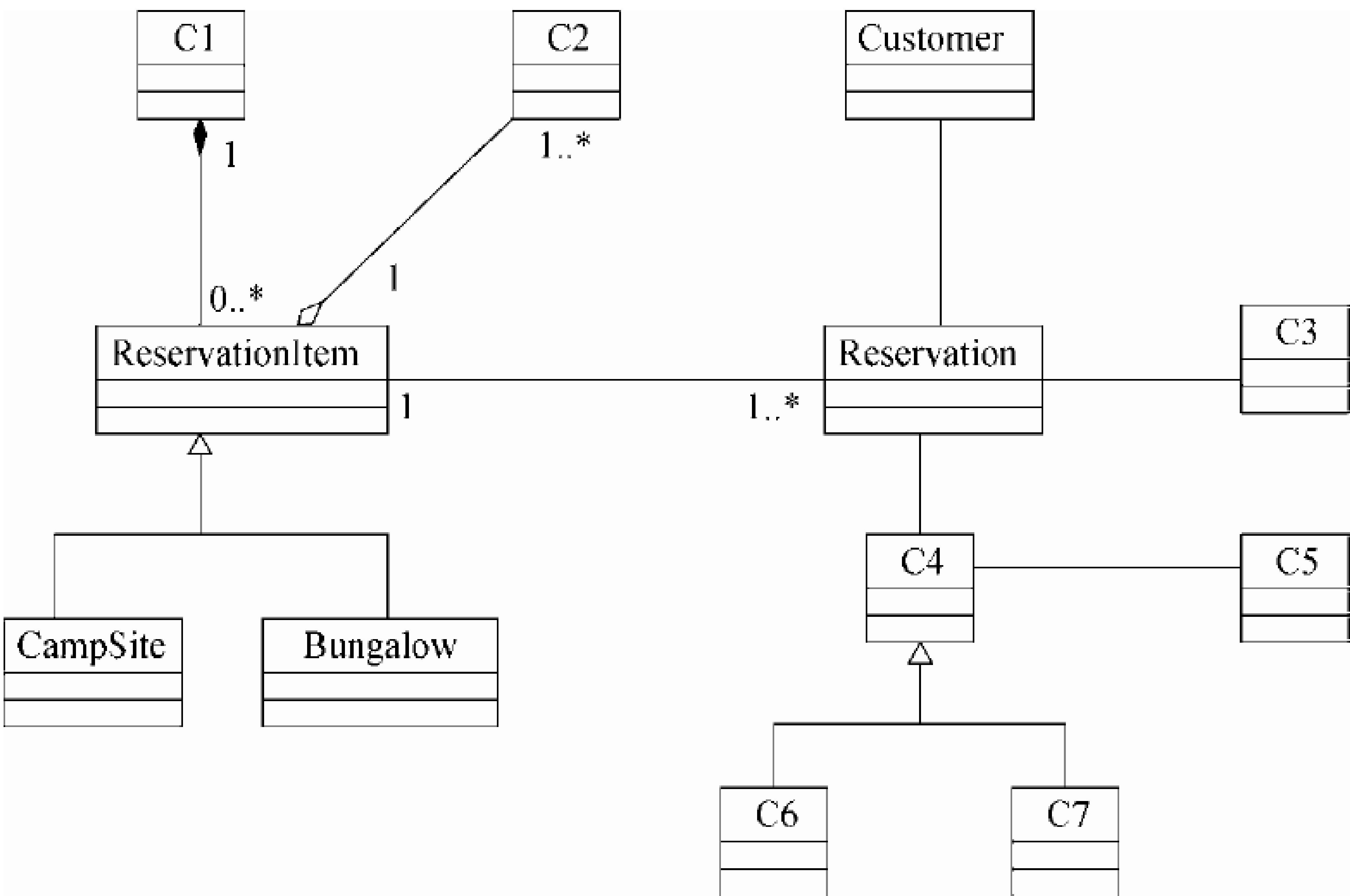


图 3-2　类图

【问题 1】（6 分）

根据说明中的描述与表 3-1，给出图 3-1 中 UC1～UC6 处所对应的用例名称。

【问题 2】（7 分）

根据说明中的描述与表 3-2，给出图 3-2 中 C1～C7 处所对应的类名。

【问题 3】（2 分）

对于某些需求量非常大的小木屋或营地，说明中功能 4 的赔偿金计算规则，不足以弥补取消预定所带来的损失。如果要根据预定的时段以及所预定场地的需求量，设计不同层次的赔偿金计算规则，需要对图 3-2 进行怎样的修改？（请用文字说明）

试题三分析

本题属于经典的考题，主要考查面向对象分析方法以及 UML 的用例图和类图的应用。

【问题 1】

本题要求将图 3-1 所给出的用例图补充完整。题目说明中已经给出了所有可能的用例的列表（如表 3-1 所示）。这就省去了寻找用例的步骤，只需要依据用例列表中给出的用例，在说明中确定用例与 Actor 之间的关系即可将图补充完整。

用例图的构成要素有：参与者（Actor）、用例（Usecase）以及用例之间的关系。题目中的信息系统的主要用户是售票处的工作人员（Ticketing Officer），所以在图 3-1 中只给出了 1 个参与者。由说明可知，售票处工作人员利用该系统可以实现 6 个主要的功能：管理预定申请（ManageInquiries）、预定（MakeReservation）、支付管理（ManagePayment）、游客取消预定（CancelReservation）、自动取消预定（AutoCancelReservation）和信息查询（CheckAvailability）。其中“管理预定申请”“支付管理”“游客取消预定”“自动取消预定”和“支付管理”均已经出现在图 3-1 中。支付租赁费用是预定过程中的一个必要步骤，而 UC2 和“支付管理”之间又是“include”关系，可以推断出 UC2 应该对应用例“预定（MakeReservation）”。那么用例“管理预定申请”和“预定”具有的相同步骤就是 UC1 所对应的用例，由此推断出 UC1 对应用例“信息查询（CheckAvailability）”。

由功能“支付管理”的说明可知，它具备两个能力：管理支付方式（信用卡或现金）以及计算折扣。UC4 和 UC5 与用例“支付管理”之间是概括关系，说明 UC4 和 UC5 是支付方式的两个特化，所以 UC4 为“现金支付（MangeCashPayment）”，UC5 为“信用卡支付（ManageCrCardPayment）”。UC3 对应“计算折扣（GetDiscount）”。

这时用例列表中只剩下用例 CalcuateRefund（计算取消预定的赔偿金）没有出现在图中了，那么它就是 UC6 对应的用例。从图 3-1 来看，UC6 应该表示用例“游客取消预定（CancelReservation）”和“自动取消预定（AutoCancelReservation）”中包含的公共事件流。不管是哪种类型的取消预定，都需要计算赔偿金，以决定退还给用户的费用，所以 UC6 对应用例 CalcuateRefund。

【问题 2】

本问题考查的是类图建模。题目中已经给出了类的列表，要求考生根据说明指出每个类在类图中的位置。在解题时，可以同时参考用例图中给出的信息。

先整体地看一下类图，寻找其中是否包含继承、聚集或组装等这些层次结构，这是快速确定部分类的关键。在图 3-2 中有一个继承结构：C4、C6 和 C7。在图 3-1 中，用例之间也有一个概括的关系，这就提示我们，C4、C6 和 C7 这 3 个类一定与支付功能相关。在表 3-2 中寻找与支付功能相关的类：Payment、CashPayment 和 CreditCardPayment。下来就是确定这 3 个类中，哪个是父类。很明显，Payment 应该作为父类。因此 C4 对应 Payment，C6 对应 CashPayment，C7 对应 CreditCardPayment（C6 和 C7 可以互换）。支付管理中还有一项计算折扣的能力，类列表中的类 Discount 表示付款折扣，而与 C5 与 C4 之间具有关联关系，所以 C5 应该对应类 Discout。

C1、C2 分别与类“ReservationItem”之间具有组装和聚集的关系，而从说明中可知，具有这种整体部分关系的只有公园、预定及租赁费用之间，所以 C1 对应 NationalPark，C2 对应 Rate。最后的一个类 C3 对应 TicketingOfficer，即用例图中的 Actor。

【问题 3】

在面向对象方法中，好的类模型对需求的变化应该具有一定的适应性。本题考察的就是这一点。根据题目，现在对原有的赔偿金计算规则要进行修正。除了考虑取消预定的时间之外，同时要考虑所预定的小木屋或营地的地段以及需求量。修正类模型时通常两种基本方式，一种是修改已有的类，使其适应新的需求；第二种是增加一个新的类来完成新的需求，但是需要同时考虑新增加的类与已有类之间的关系。这道题目两种修改方法都可以采用。

若要修改已有的类，需要首先了解哪个类与现在的新需求是有相关性的。新需求针对的是赔偿金，赔偿金又与租赁费用相关，所以要找原先与租赁费用相关的那个类，即 Rate。解决方案之一就是修改 Rate，使其能够按照新的规则计算赔偿金。

第二种修改方式，增加一个专门计算赔偿金的类。按照新的计算规则，这个类就与游客的每次预定内容相关，因此这个新增加的类应该与类 ReservationItem 之间有关联关系。

参考答案

【问题 1】

UC1：CheckAvailablity　　UC2：MakeReservation

UC3：GetDiscount　　UC4：ManageCashPayment

UC5：ManageCrCardPayment　　UC6：CalculateRefund

UC4 和 UC5 可以互换。

【问题 2】

C1：NationalPark　　C2：Rate　　C3：TicketingOfficer

C4：Payment　　C5：Discount　　C6：CashPayment

C7：CreditCardPayment

C6 和 C7 可以互换。

【问题 3】

解答 1：增加一个新的类，该类与类 ReservationItem 之间有关联关系。

或

解答 2：修改 Rate 类，使其具有计算赔偿金的功能。

试题四（共 15 分）

阅读下列说明和 C 代码，回答问题 1 至问题 3，将解答写在答题纸的对应栏内。

【说明】

设有 n 个货物要装入若干个容量为 C 的集装箱以便运输，这 n 个货物的体积分别为 $\{s_1,s_2,\cdots,s_n\}$，且有 $s_i \leqslant C(1 \leqslant i \leqslant n)$。为节省运输成本，用尽可能少的集装箱来装运这 n 个货物。

下面分别采用最先适宜策略和最优适宜策略来求解该问题。

最先适宜策略（firstfit）首先将所有的集装箱初始化为空，对于所有货物，按照所给的次序，每次将一个货物装入第一个能容纳它的集装箱中。

最优适宜策略（bestfit）与最先适宜策略类似，不同的是，总是把货物装到能容纳它且目前剩余容量最小的集装箱，使得该箱子装入货物后闲置空间最小。

【C 代码】

下面是这两个算法的 C 语言核心代码。

（1）变量说明

n：货物数

C：集装箱容量

s：数组，长度为 n，其中每个元素表示货物的体积，下标从 0 开始

b：数组，长度为 n，b[i]表示第 i+1 个集装箱当前已经装入货物的体积，下标从 0 开始

i, j：循环变量

k：所需的集装箱数

min：当前所用的各集装箱装入了第 i 个货物后的最小剩余容量

m：当前所需要的集装箱数

temp：临时变量

（2）函数 firstfit

```
int firstfit() {
        int i, j;
        k = 0;
        for(i = 0; i < n; i++) {
```

```
        b[i] = 0;
      }
      for(i = 0; i < n; i++){
        ____(1)____;
        while(C - b[j] < s[i]){
          j++;
        }
        ____(2)____;
        k = k > (j+1) ? k : (j+1) ;
      }
      return k;
}
```

（3）函数 bestfit

```
int bestfit()  {
      int i, j, min, m, temp;
      k = 0;
      for(i = 0; i < n; i++) {
        b[i] = 0;
      }
      for(i = 0; i < n; i++) {
        min = C;
        m = k + 1;
        for(j = 0; j < k + 1; j++) {
          temp = C - b[j] - s[i];
          if(temp > 0 && temp < min){
            ____(3)____;
            m = j;
          }
        }
        ____(4)____;
        k = k > (m+1) ? k : (m+1) ;
      }
      return k;
}
```

【问题 1】（8 分）

根据说明和 C 代码，填充 C 代码中的空（1）～（4）。

【问题 2】（4 分）

根据说明和 C 代码，该问题在最先适宜和最优适宜策略下分别采用了__（5）__和__（6）__

算法设计策略，时间复杂度分别为____（7）____和____（8）____（用 O 符号表示）。

【问题 3】（3 分）

考虑实例 n = 10，C = 10，各个货物的体积为{4,2,7,3,5,4,2,3,6,2}。该实例在最先适宜和最优适宜策略下所需的集装箱数分别为____（9）____和____（10）____。考虑一般的情况，这两种求解策略能否确保得到最优解？____（11）____（能或否）

试题四分析

本题考查算法设计与分析。

【问题 1】

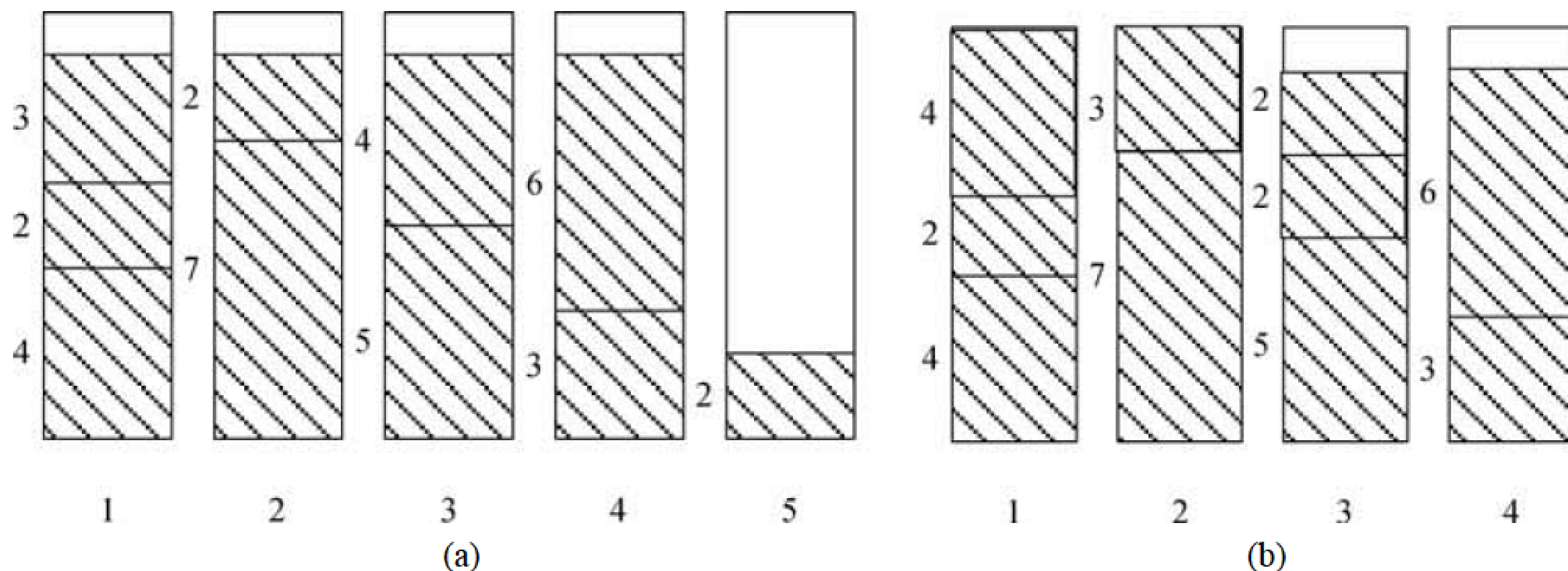
根据最先适宜算法思想，每取出一个货物，从第一个集装箱开始判断该货物是否能放入集装箱，若能则放入，因此空(1)填 j = 0。while 循环判断，若货物不能放入集装箱，则考虑下一个集装箱。不满足 while 循环中的条件，说明货物能放入集装箱，因此空(2)填 b[j] = b[j] + s[i]。根据最优适宜算法思想，每取出一个货物，从第一个集装箱开始，确定能放入该货物且剩余容量最小的集装箱，并把该货物放入该集装箱中。if 条件判断，若找到了比能放入货物且剩余容量更小的集装箱，则剩余容量最小值改为当前的集装箱的剩余容量，因此空(3)填 min = temp。确定了集装箱后，把货物装入集装箱中，空(4) b[m] = b[m] + s[i]。

【问题 2】

最先适宜算法总是把货物放入第一个能放入的集装箱，最优适宜算法总是把货物放入能容纳该货物且剩余容量最小的集装箱，因此都是基于贪心策略进行的，空(5)和(6)填贪心。函数 firstfit 中的 for 循环考虑 n 个货物，其中嵌套了 while 循环，最多的集装箱数为 n，因此时间复杂度为 $O(n^2)$。函数 bestfit 中的 for 循环考虑 n 个货物，其中嵌套了 for 循环检查每个集装箱的剩余容量，最多的集装箱数为 n，因此时间复杂度为 $O(n^2)$。

【问题 3】

对实例 n = 10, C = 10, S = {4,2,7,3,5,4,2,3,6,2}，根据最先适宜和最优适宜算法，其具体的装箱方案分别如下图(a)和(b)所示。



(a) (b)

因此最先适宜和最优适宜方法所需的集装箱数分别为 5 和 4，装箱问题是一个非常难的问题，这两种贪心策略不能确保得到最优解，即最少的装箱数。

参考答案

【问题 1】

（1）j = 0

（2）b[j] = b[j] + s[i]　　及其等价形式

（3）min = temp

（4）b[m] = b[m] + s[i]　及其等价形式

【问题 2】

（5）贪心　　（6）贪心　　（7）$O(n^2)$　　（8）$O(n^2)$

【问题 3】

（9）5　　（10）4　　（11）否

试题五（共 15 分）

阅读下列说明和 C++代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

现欲开发一个软件系统，要求能够同时支持多种不同的数据库，为此采用抽象工厂模式设计该系统。以 SQL Server 和 Access 两种数据库以及系统中的数据库表 Department 为例，其类图如图 5-1 所示。

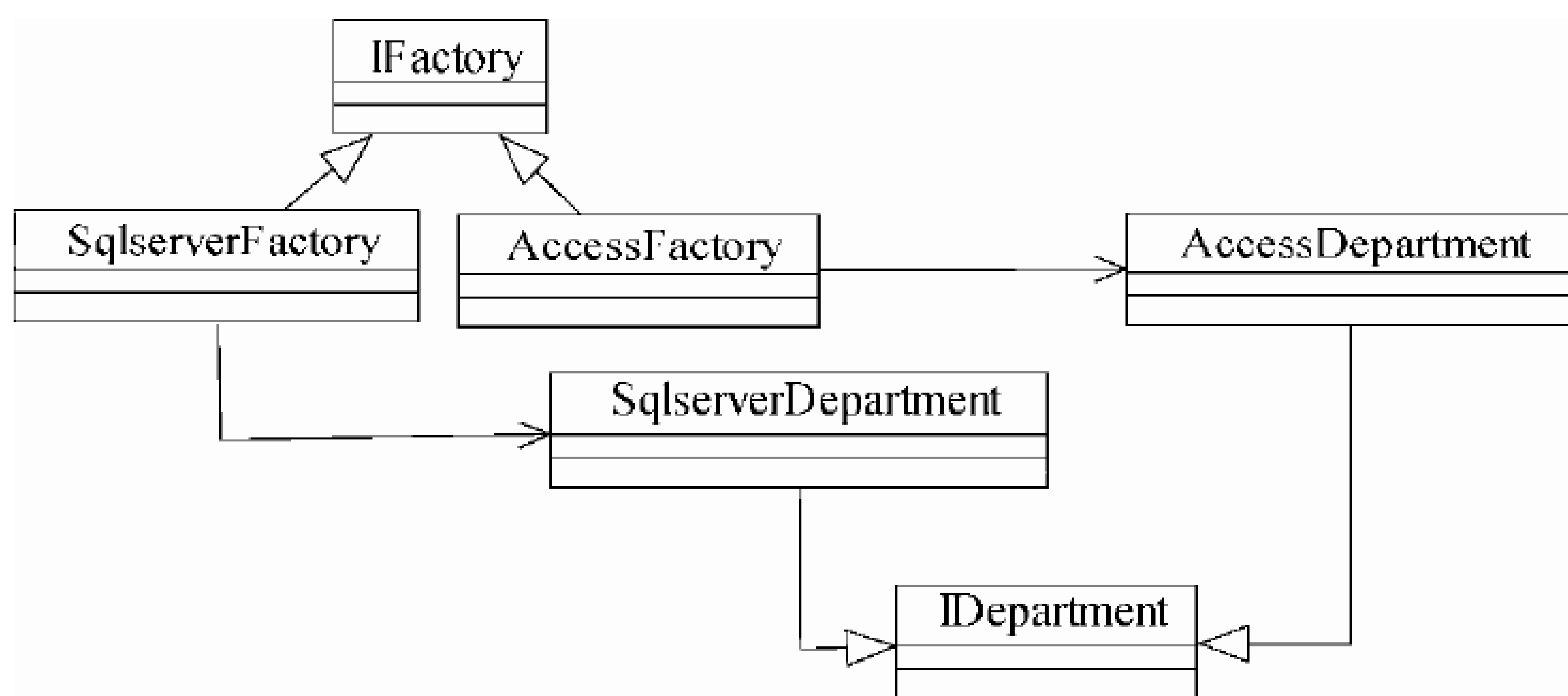


图 5-1　类图

【C++代码】

```
#include <iostream>
using namespace std;

class Department { /* 代码省略 */ };
class IDepartment {
public:
```

```
    ____(1)____ = 0;
    ____(2)____ = 0;
};

class SqlserverDepartment : ____(3)____ {
public:
    void Insert(Department* department) {
        cout << "Insert a record into Department in SQL Server!\n";
// 其余代码省略
}
    Department GetDepartment(int id) {
        /* 代码省略 */
}
};

class AccessDepartment : ____(4)____{
public:
    void Insert(Department* department) {
        cout << "Insert a record into Department in ACCESS!\n";
        // 其余代码省略
}
    Department GetDepartment(int id) {
        /* 代码省略 */
}
};

____(5)____ {
public:
    ____(6)____ = 0;
};

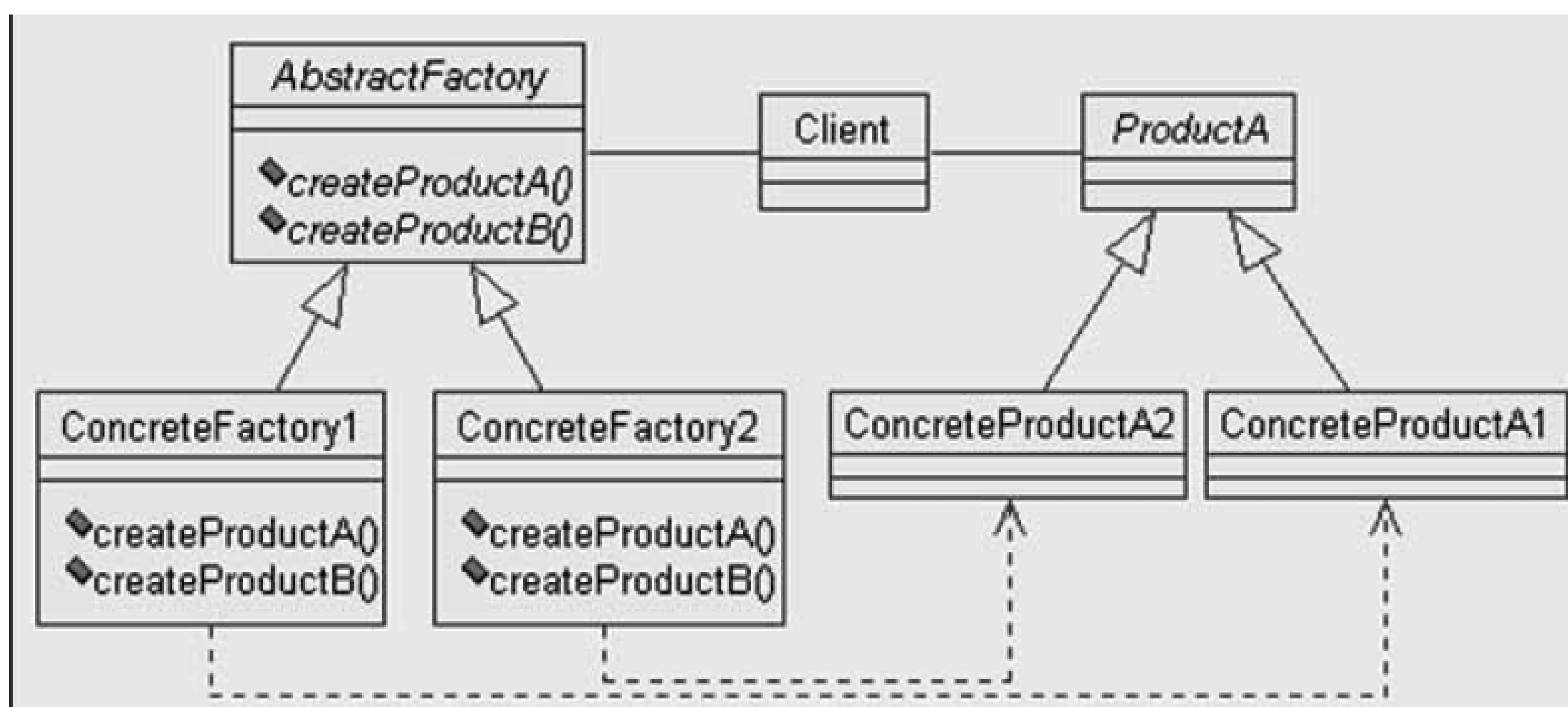
class SqlServerFactory : public IFactory {
public:
    IDepartment* CreateDepartment() { return new SqlserverDepartment(); }
// 其余代码省略
};

class AccessFactory : public IFactory {
public:
    IDepartment* CreateDepartment() { return new AccessDepartment(); }
// 其余代码省略
};
```

试题五分析

本题考查抽象工厂（Abstract Factory）模式的概念及应用。

Abstract Factory 模式的意图是，提供一个创建一系列相关或相互依赖对象的接口，而无须指定它们具体的类。Abstract Factor 模式的结构如下图所示。



其中，类 AbstractFactory 声明一个创建抽象产品对象的操作接口；类 ConcreteFactory 实现创建具体产品对象的操作；类 ProductA 为一类产品对象声明一个接口；类 ConcreteProduct 具有两个功能：定义一个将被相应的具体工厂创建的产品对象；实现 ProductA 接口；类 Client 仅使用由 AbstractFactory 和 AbstractProduct 类声明的接口。

在以下情况可以使用 AbstractFactory 模式：

（1）一个系统要独立于它的产品的创建、组合和表示时；

（2）一个系统要由多个产品系统中的一个来配置时；

（3）当要强调一系列相关的产品对象的设计一边进行联合使用时；

（4）提供一个产品类库，而只想显示它们的接口而不是实现时。

题目利用抽象工厂模式来解决在同一个软件系统中支持多种不同数据库的问题，这也是软件开发中比较常见的情形。其中的类 IFactory 相当于上图中的类 AbstractFactory；类 IDepartment 相当于上图中的类 ProdcutA。本题中只给出了一个产品类。

下面来分析程序。

第（1）、（2）空出现类 IDepartment 的定义中。类 IDepartment 的作用是为一类产品对象声明一个接口，在 C++中通常都采用抽象类来定义这种抽象操作接口。C++中的抽象类是包含了至少一个纯虚拟函数的类。纯虚拟函数的语法是：

virtual <函数名>(<参数列表>) =0;

在程序中已经出现了纯虚拟函数的标志“=0”，因此（1）、（2）空应该都是纯虚拟函数。下面来确定纯虚拟函数的原型。这需要去考察类 IDepartment 的子类，因此纯虚拟函数是在父类中定义，在子类中实现。由类图 5-1 可知，类 IDepartment 的子类分别是 SqlserverDepartment 和 AccessDepartment。至此可以提前确定（3）、（4）空的内容了，即其所对应的父类。因此（3）、（4）空都应该填写 public IDepartment。我们看类

SqlserverDepartment 中的方法，分别为：Insert 和 GetDepartment，而在类 AccessDepartment 中也出现了这两个方法，且接口完全一致。所以这两个方法就应该是类 IDepartment 中所定义的抽象接口。由此可知，（1）空应该填写 virtual void Insert(Department* department)，（2）空应该填写 virtual Department GetDepartment(int id)。

空（5）和（6）分别缺失在类的名称以及该类中的方法。由图 5-1 和代码可知，缺少类 IFactory 的定义，所以（5）空处应该填写 class IFactory。那么类 IFactory 应包含的方法是什么？类 IFactory 的作用是声明一个创建抽象产品对象的操作接口，这个接口一定会同时出现在 IFactory 的子类 SqlServerFactory 和 AccessFactory 中，即 CreateDepartment。（6）处同样应该是一个纯虚拟函数，所以（6）空处应该填写 virtual IDepartment* CreateDepartment()。

参考答案

（1）virtual void Insert(Department* department)

（2）virtual Department GetDepartment(int id)

（3）public IDepartment

（4）public IDepartment

（5）class IFactory

（6）virtual IDepartment* CreateDepartment()

试题六（共 15 分）

阅读下列说明和 Java 代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

现欲开发一个软件系统，要求能够同时支持多种不同的数据库，为此采用抽象工厂模式设计该系统。以 SQL Server 和 Access 两种数据库以及系统中的数据库表 Department 为例，其类图如图 6-1 所示。

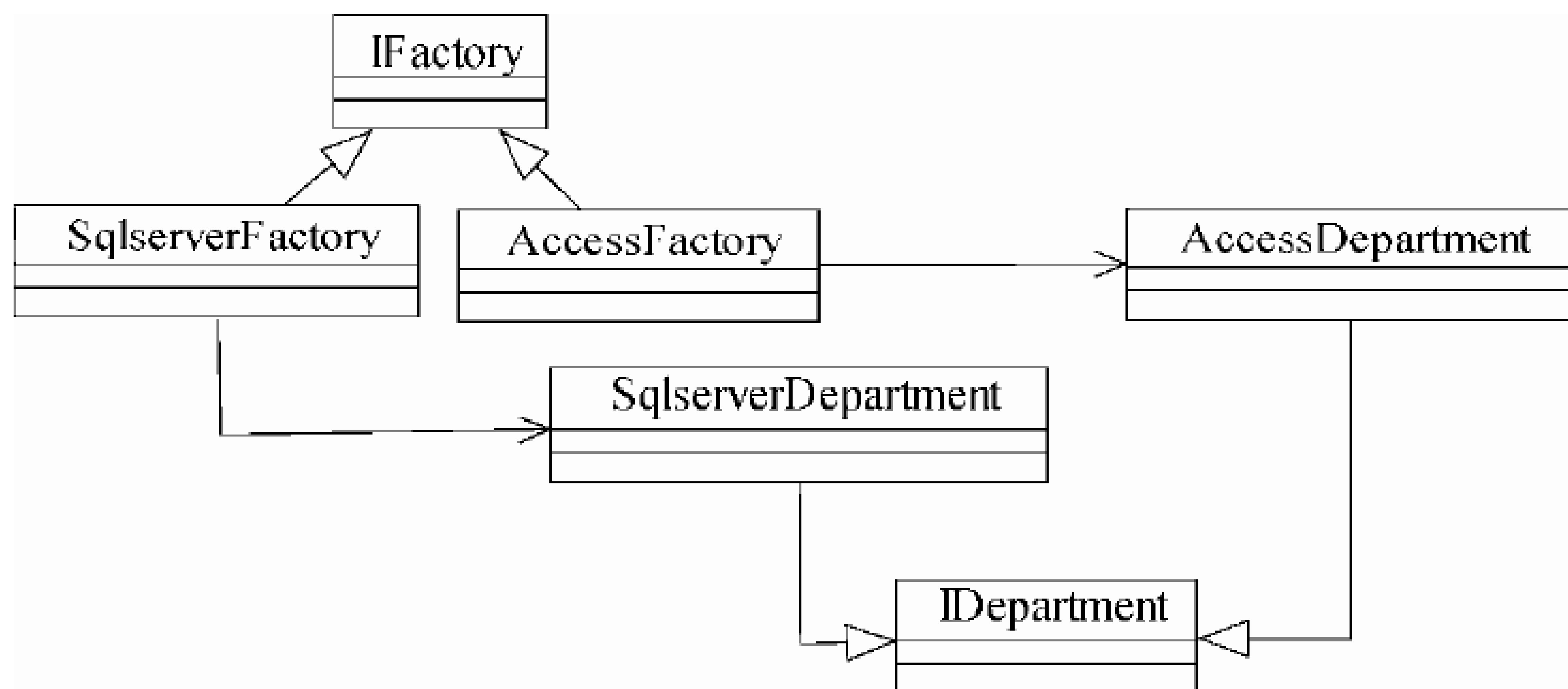


图 6-1　类图

【Java 代码】

```
import java.util.*;
```

```
class Department {  /* 代码省略 */   }

interface IDepartment {
    ______(1)______;
    ______(2)______;
}

class SqlserverDepartment ______(3)______ {
    public void Insert(Department department) {
        System.out.println("Insert a record into Department in SQL Server!");
        // 其余代码省略
    }
    public Department GetDepartment(int id) {
/*  代码省略 */
    }
}

class AccessDepartment ______(4)______ {
    public void Insert(Department department) {
        System.out.println("Insert a record into Department in ACCESS!");
        // 其余代码省略
    }
    public Department GetDepartment(int id) {
/*  代码省略   */
    }
}

______(5)______ {
    ______(6)______;
}

class SqlServerFactory implements IFactory {
    public IDepartment CreateDepartment() {
        return new SqlserverDepartment();
    }
    // 其余代码省略
}

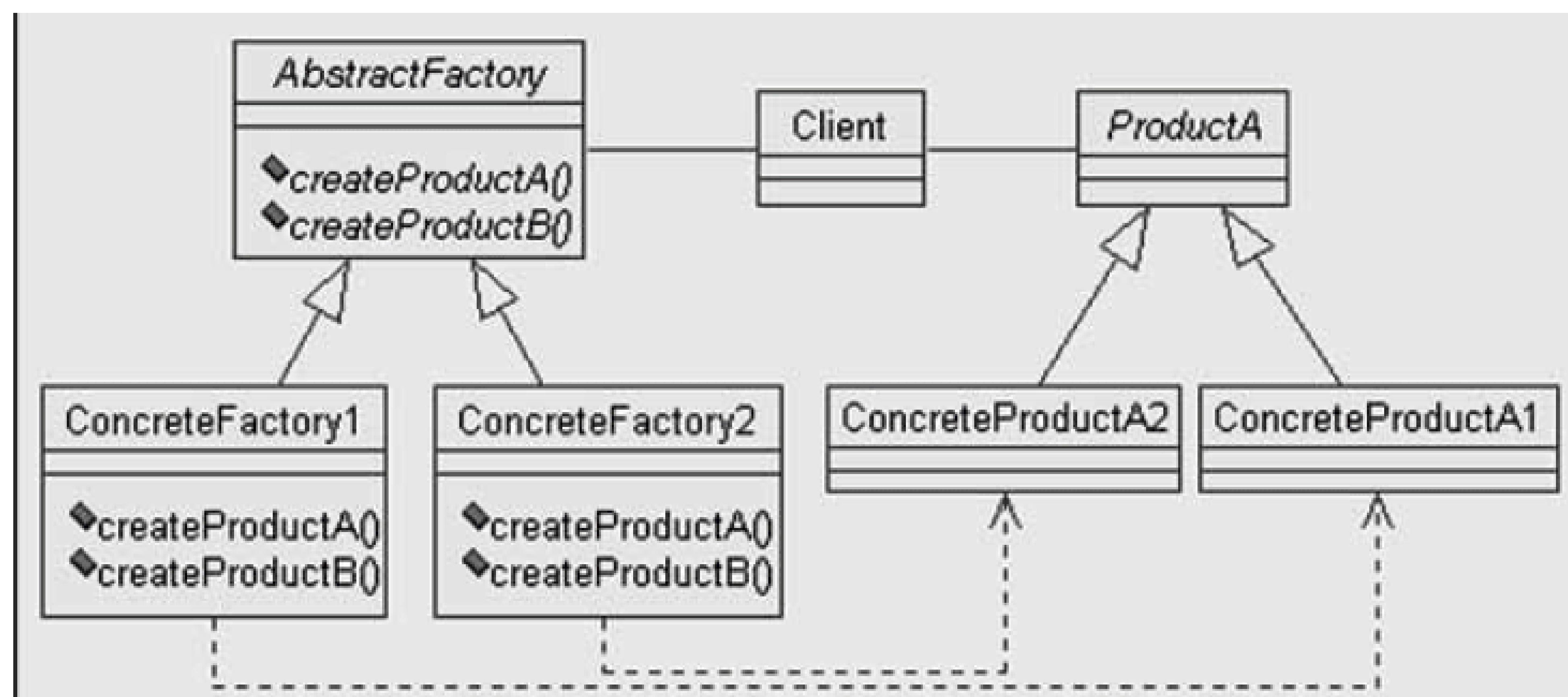
class AccessFactory implements IFactory {
    public IDepartment CreateDepartment() {
        return new AccessDepartment();
```

```
        }
        // 其余代码省略
    }
```

试题六分析

本题考查抽象工厂（Abstract Factory）模式的概念及应用。

Abstract Factory 模式的意图是，提供一个创建一系列相关或相互依赖对象的接口，而无须指定它们具体的类。Abstract Factor 模式的结构如下图所示。



其中，类 AbstractFactory 声明一个创建抽象产品对象的操作接口；类 ConcreteFactory 实现创建具体产品对象的操作；类 ProductA 为一类产品对象声明一个接口；类 ConcreteProduct 具有两个功能：定义一个将被相应的具体工厂创建的产品对象；实现 ProductA 接口；类 Client 仅使用由 AbstractFactory 和 AbstractProduct 类声明的接口。

在以下情况可以使用 Abstract Factory 模式：

（1）一个系统要独立于它的产品的创建、组合和表示时；

（2）一个系统要由多个产品系统中的一个来配置时；

（3）当要强调一系列相关的产品对象的设计一边进行联合使用时；

（4）提供一个产品类库，而只想显示它们的接口而不是实现时。

题目利用抽象工厂模式来解决在同一个软件系统中支持多种不同数据库的问题，这也是软件开发中比较常见的情形。其中的类 IFactory 相当于上图中的类 AbstractFactory；类 IDepartment 相当于上图中的类 ProdcutA。本题中只给出了一个产品类。

下面来分析程序。

第（1）、（2）空出现类 IDepartment 的定义中。类 IDepartment 的作用是为一类产品对象声明一个接口，在 Java 中通常都采用 Inteface 来定义这种抽象操作接口。要确定接口的原型，需要去考察实现接口 IDepartment 的类。由类图 6-1 可知，实现接口 IDepartment 的类分别是 SqlserverDepartment 和 AccessDepartment。至此可以提前确定（3）、（4）空的内容了。（3）、（4）空都应该填写 implements IDepartment。我们看类 SqlserverDepartment 中的方法，分别为：Insert 和 GetDepartment，而在类 AccessDepartment 中也出现了这两个方法，且接口完全一致。所以这两个方法就应该是 IDepartment 中所定义的抽象接口。由

此可知，（1）空应该填写 void Insert(Department department)，（2）空应该填写 Department GetDepartment(int id)。

空（5）和（6）分别缺失在类的名称以及该类中的方法。由图 6-1 和代码可知，缺少类 IFactory 的定义。由代码“class SqlServerFactory implements IFactory”和“class AccessFactory implements IFactory”可知，IFactory 也是一个接口。所以（5）空处应该填写 Interface IFactory。那么类 IFactory 应包含的方法是什么？类 IFactory 的作用是声明一个创建抽象产品对象的操作接口，这个接口一定会同时出现在类 SqlServerFactory 和 AccessFactory 中，即 CreateDepartment。（6）处同样应该是一个纯虚拟函数，所以（6）空处应该填写 IDepartment CreateDepartment()。

参考答案

（1）void Insert(Department department)

（2）Department GetDepartment(int id)

（3）implements IDepartment

（4）implements IDepartment

（5）interface IFactory

（6）IDepartment CreateDepartment()

第 17 章　2013 上半年软件设计师上午试题分析与解答

试题（1）

常用的虚拟存储器由__（1）__两级存储器组成。

（1）A．主存-辅存　　B．主存-网盘　　C．Cache-主存　　D．Cache-硬盘

试题（1）分析

本题考查计算机系统存储系统基础知识。

在具有层次结构存储器的计算机中，虚拟存储器是为用户提供一个比主存储器大得多的可随机访问的地址空间的技术。虚拟存储技术使辅助存储器和主存储器密切配合，对用户来说，好像计算机具有一个容量比实际主存大得多的主存可供使用，因此称为虚拟存储器。虚拟存储器的地址称为虚地址或逻辑地址。

参考答案

（1）A

试题（2）

中断向量可提供__（2）__。

（2）A．I/O 设备的端口地址　　B．所传送数据的起始地址
　　C．中断服务程序的入口地址　　D．主程序的断点地址

试题（2）分析

本题考查计算机系统基础知识。

计算机在执行程序过程中，当遇到急需处理的事件时，暂停当前正在运行的程序，转去执行有关服务程序，处理完后自动返回源程序，这个过程称为中断。

中断是一种非常重要的技术，输入输出设备和主机交换数据、分时操作、实时系统、计算机网络和分布式计算机系统中都要用到这种技术。为了提高响应中断的速度，通常把所有中断服务程序的入口地址（或称为中断向量）汇集为中断向量表。

参考答案

（2）C

试题（3）

为了便于实现多级中断嵌套，使用__（3）__来保护断点和现场最有效。

（3）A．ROM　　B．中断向量表　　C．通用寄存器　　D．堆栈

试题（3）分析

本题考查计算机系统基础知识。

当系统中有多个中断请求时，中断系统按优先级进行排队。若在处理低级中断过程

中又有高级中断申请中断，则高级中断可以打断低级中断处理，转去处理高级中断，等处理完高级中断后再返回去处理原来的低级中断，称为中断嵌套。实现中断嵌套用后进先出的栈来保护断点和现场最有效。

参考答案

（3）D

试题（4）

DMA 工作方式下，在__（4）__之间建立了直接的数据通路。

（4）A．CPU 与外设　B．CPU 与主存　C．主存与外设　D．外设与外设

试题（4）分析

本题考查计算机系统基础知识。

计算机系统中主机与外设间的输入输出控制方式有多种，在 DMA 方式下，输入输出设备与内存储器直接相连，数据传送由 DMA 控制器而不是主机 CPU 控制。CPU 除了传送开始和终了时进行必要的处理外，不参与数据传送的过程。

参考答案

（4）C

试题（5）、（6）

地址编号从 80000H 到 BFFFFH 且按字节编址的内存容量为__（5）__KB，若用 16K×4bit 的存储器芯片构成该内存，共需__（6）__片。

（5）A．128　B．256　C．512　D．1024

（6）A．8　B．16　C．32　D．64

试题（5）、（6）分析

本题考查计算机系统基础知识。

从 80000H 到 BFFFFH 的编址单元共 3FFFF（即 2^{18}）个，按字节编址的话，对应的容量为 2^8 KB，即 256KB。若用 16K×4bit 的芯片构成该内存，构成一个 16KB 存储器需要 2 片，256÷16=16，因此共需要 32 片。

参考答案

（5）B　（6）C

试题（7）

利用报文摘要算法生成报文摘要的目的是__（7）__。

（7）A．验证通信对方的身份，防止假冒

B．对传输数据进行加密，防止数据被窃听

C．防止发送方否认发送过的数据

D．防止发送的报文被篡改

试题（7）分析

本题考查报文摘要的知识。报文摘要是指单向哈希函数算法将任意长度的输入报文

经计算得出固定位的输出。报文摘要是用来保证数据完整性的。传输的数据一旦被修改，那么计算出的摘要就不同，只要对比两次摘要就可确定数据是否被修改过。

参考答案

（7）D

试题（8）

防火墙通常分为内网、外网和 DMZ 三个区域，按照受保护程度，从高到低正确的排列次序为<u>（8）</u>。

（8）A．内网、外网和 DMZ B．外网、内网和 DMZ
C．DMZ、内网和外网 D．内网、DMZ 和外网

试题（8）分析

本题考查防火墙的基础知识。

通过防火墙我们可以将网络划分为三个区域：安全级别最高的 LAN Area（内网），安全级别中等的 DMZ 区域和安全级别最低的 Internet 区域（外网）。三个区域因担负不同的任务而拥有不同的访问策略。通常的规则如下：

① 内网可以访问外网：内网的用户需要自由地访问外网。在这一策略中，防火墙需要执行 NAT。

② 内网可以访问 DMZ：此策略使内网用户可以使用或者管理 DMZ 中的服务器。

③ 外网不能访问内网：这是防火墙的基本策略，内网中存放的是公司内部数据，显然这些数据是不允许外网的用户进行访问的。如果要访问，就要通过 VPN 方式来进行。

④ 外网可以访问 DMZ：DMZ 中的服务器需要为外界提供服务，所以外网必须可以访问 DMZ。同时，外网访问 DMZ 需要由防火墙完成对外地址到服务器实际地址的转换。

⑤ DMZ 不能访问内网：如不执行此策略，则当入侵者攻陷 DMZ 时，内部网络将不会受保护。

⑥ DMZ 不能访问外网：此条策略也有例外，可以根据需要设定某个特定的服务器可以访问外网，以保证该服务器可以正常工作。

综上所述，防火墙区域按照受保护程度从高到低正确的排列次序应为内网、DMZ 和外网。

参考答案

（8）D

试题（9）

近年来，在我国出现的各类病毒中，<u>（9）</u>病毒通过木马形式感染智能手机。

（9）A．欢乐时光 B．熊猫烧香 C．X 卧底 D．CIH

试题（9）分析

本题考查病毒及其危害。

欢乐时光及熊猫烧香均为蠕虫病毒，CIH 则为系统病毒，这 3 者均以感染台式机或服务器为主，且产生较早；X 卧底则是新近产生的、通过木马形式传播、目标为智能手机的病毒。

参考答案

（9）C

试题（10）

王某是一名软件设计师，按公司规定编写软件文档，并上交公司存档。这些软件文档属于职务作品，且__（10）__。

（10）A. 其著作权由公司享有

B. 其著作权由软件设计师享有

C. 除其署名权以外，著作权的其他权利由软件设计师享有

D. 其著作权由公司和软件设计师共同享有

试题（10）分析

本题考查知识产权知识。公民为完成法人或者其他组织工作任务所创作的作品是职务作品。职务作品可以是作品分类中的任何一种形式，如文字作品、电影作品、计算机软件等。职务作品的著作权归属分两种情形。

一般职务作品的著作权由作者享有。所谓一般职务作品是指虽是为完成工作任务而为，但非经法人或其他组织主持，不代表其意志创作，也不由其承担责任的职务作品。对于一般职务作品，法人或其他组织享有在其业务范围内优先使用的权利，期限为两年。优先使用权是专有的，未经单位同意，作者不得许可第三人以与法人或其他组织使用的相同方式使用该作品。在作品完成两年内，如单位在其业务范围内不使用，作者可以要求单位同意由第三人以与法人或其他组织使用的相同方式使用，所获报酬，由作者与单位按约定的比例分配。

特殊的职务作品，除署名权以外，著作权的其他权利由法人或者其他组织（单位）享有。所谓特殊职务作品是指著作权法第 16 条第 2 款规定的两种情况：一是主要利用法人或者其他组织的物质技术条件创作，并由法人或者其他组织承担责任的工程设计、产品设计图、计算机软件、地图等科学技术作品；二是法律、法规规定或合同约定著作权由单位享有的职务作品。

参考答案：

（10）A

试题（11）

甲经销商擅自复制并销售乙公司开发的 OA 软件光盘已构成侵权。丙企业在未知的情形下从甲经销商处购入 10 张并已安装使用。在丙企业知道了所使用的软件为侵权复制品的情形下，以下说法正确的是__（11）__。

（11）A. 丙企业的使用行为侵权，须承担赔偿责任
B. 丙企业的使用行为不侵权，可以继续使用这 10 张软件光盘
C. 丙企业的使用行为侵权，支付合理费用后可以继续使用这 10 张软件光盘
D. 丙企业的使用行为不侵权，不需承担任何法律责任

试题（11）分析

本题考查知识产权知识。我国计算机软件保护条例第 30 条规定“软件的复制品持有人不知道也没有合理理由应当知道该软件是侵权复制品的，不承担赔偿责任；但是，应当停止使用、销毁该侵权复制品。如果停止使用并销毁该侵权复制品将给复制品使用人造成重大损失的，复制品使用人可以在向软件著作权人支付合理费用后继续使用。”丙企业在获得软件复制品的形式上是合法的（向经销商购买），但是由于其没有得到真正软件权利人的授权，其取得的复制品仍是非法的，所以丙企业的使用行为属于侵权行为。

丙企业应当承担的法律责任种类和划分根据主观状态来确定。首先，法律确立了软件著作权人的权利进行绝对的保护原则，即软件复制品持有人不知道也没有合理理由应当知道该软件是侵权复制品的，也必须承担停止侵害的法律责任，只是在停止使用并销毁该侵权复制品将给复制品使用人造成重大损失的情况下，软件复制品使用人可继续使用，但前提是必须向软件著作权人支付合理费用。其次，如果软件复制品持有人能够证明自己确实不知道并且也没有合理理由应当知道该软件是侵权复制品的，软件复制品持有人除承担停止侵害外，不承担赔偿责任。

软件复制品持有人一旦知道了所使用的软件为侵权复制品时，应当履行停止使用、销毁该软件的义务。不履行该义务，软件著作权人可以诉请法院判决停止使用并销毁侵权软件。如果软件复制品持有人在知道所持有软件是非法复制品后继续使用给权利人造成损失的，应该承担赔偿责任。

参考答案

（11）C

试题（12）

声音信号数字化过程中首先要进行__（12）__。

（12）A. 解码 B. D/A 转换 C. 编码 D. A/D 转换

试题（12）分析

本题考查多媒体基础知识。声音信号是一种模拟信号，计算机要对它进行处理，必须将它转换成为数字声音信号，即用二进制数字的编码形式来表示声音，通常将这一过程称为数字化过程。声音信号数字化过程中首先是将模拟信号转换成离散信号，即 A/D 转换（模数转换）。

参考答案

（12）D

试题（13）

以下关于 dpi 的叙述中，正确的是__（13）__。

（13）A．每英寸的 bit 数　　B．存储每个像素所用的位数

C．每英寸像素点　　D．显示屏上能够显示出的像素数目

试题（13）分析

本题考查多媒体基础知识。dpi 是描述图像分辨率的单位，表示每英寸多少像素点，即组成一幅图像的像素密度。它实质上是图像数字化的采样间隔，由它确立组成一幅图像的像素数目。对同样大小的一幅图，如果组成该图像的图像像素数目越多，则说明图像的分辨率越高，图像看起来就越逼真。相反，图像则显得越粗糙。因此，不同的分辨率会造成不同的图像清晰度。存储每个像素所用的位数是用来度量图像的分辨率的。像素深度确定彩色图像的每个像素可能有的颜色数，即确定彩色图像中可出现的最多颜色数。显示屏上能够显示出的像素数目是指显示分辨率。

参考答案

（13）C

试题（14）

媒体可以分为感觉媒体、表示媒体、表现媒体、存储媒体、传输媒体，__（14）__属于表现媒体。

（14）A．打印机　　B．硬盘　　C．光缆　　D．图像

试题（14）分析

本题考查多媒体基础知识。表现媒体指实现信息输入和输出的媒体，如键盘、鼠标、扫描仪、话筒、摄像机等为输入媒体；显示器、打印机、喇叭等为输出媒体。硬盘属于存储媒体；光缆属于传输媒体；图像属于感觉媒体。

参考答案

（14）A

试题（15）

统一过程模型是一种“用例和风险驱动，以架构为中心，迭代并且增量”的开发过程，定义了不同阶段及其制品，其中精化阶段关注__（15）__。

（15）A．项目的初创活动

B．需求分析和架构演进

C．系统的构建，产生实现模型

D．软件提交方面的工作，产生软件增量

试题（15）分析

本题考查软件开发过程模型的基本概念。统一过程模型是一种“用例和风险驱动，以架构为中心，迭代并且增量”的开发过程，由 UML 方法和工具支持，定义了不同阶段及其制品。

起始阶段专注于项目的初创活动。精化阶段理解了最初的领域范围之后，进行需求

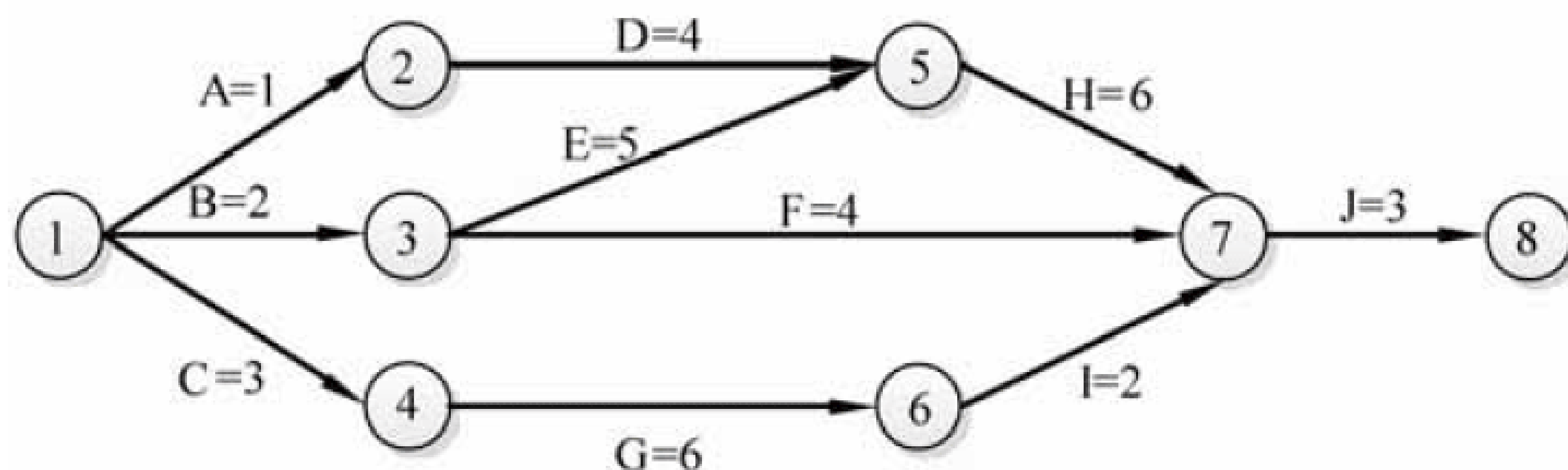
分析和架构演进。构建阶段关注系统的构建，产生实现模型。移交阶段关注于软件提交方面的工作，产生软件增量。产生阶段运行软件并监控软件的持续使用，提供运行环境的支持，提交并评估缺陷报告和变更请求。

参考答案

（15）B

试题（16）、（17）

在进行进度安排时，PERT 图不能清晰地描述__（16）__，但可以给出哪些任务完成后才能开始另一些任务。某项目 X 包含任务 A、B、……、J，其 PERT 图如下图所示（A=1 表示任务 A 的持续时间是 1 天），则项目 X 的关键路径是__（17）__。



（16）A. 每个任务从何时开始　　B. 每个任务到何时结束
C. 各任务之间的并行情况　　D. 各任务之间的依赖关系

（17）A. A-D-H-J　　B. B-E-H-J　　C. B-F-J　　D. C-G-I-J

试题（16）、（17）分析

本题考查项目管理及工具技术。

PERT 图可以清晰地表示各任务的开始时间和结束时间以及各任务之间的依赖关系，但是无法很好地表示各任务之间的并行情况。

根据关键路径法，计算出项目 X 中的关键路径为 B-E-H-J，关键路径长度为 16。

参考答案

（16）C　　（17）B

试题（18）

“软件产品必须能够在 3 秒内对用户请求作出响应”属于软件需求中的__（18）__。

（18）A. 功能需求　　B. 非功能需求　　C. 设计约束　　D. 逻辑需求

试题（18）分析

本题考查软件需求分类基础知识。

软件需求是软件系统必须完成的事以及必须具备的品质。软件需求包括功能需求、非功能需求和设计约束三个方面的内容。功能需求是所开发的软件必须具备什么样的功能；非功能需求是指产品必须具备的属性或品质，如可靠性、性能、响应时间和扩展性等等；设计约束通常对解决方案的一些约束说明。“软件产品必须能够在 3 秒内对用户请

求作出响应”主要表述软件的响应时间，属于非功能需求。

参考答案

（18）B

试题（19）

某项目为了修正一个错误而进行了修改。错误修正后，还需要进行__（19）__以发现这一修正是否引起原本正确运行的代码出错。

（19）A．单元测试 B．接受测试 C．安装测试 D．回归测试

试题（19）分析

本题考查软件测试基础知识。单元测试是在模块编写完成且无编译错误后进行，侧重于模块中的内部处理逻辑和数据结构；接受测试主要是用户为主的测试；安装测试是将软件系统安装在实际运行环境的测试；回归测试是在系统有任何修改的情况下，需要重新对整个软件系统进行的测试。

参考答案

（19）D

试题（20）

以下关于解释程序和编译程序的叙述中，正确的是__（20）__。

（20）A．编译程序和解释程序都生成源程序的目标程序

B．编译程序和解释程序都不生成源程序的目标程序

C．编译程序生成源程序的目标程序，解释程序则不然

D．编译程序不生成源程序的目标程序，而解释程序反之

试题（20）分析

本题考查程序语言翻译基础知识。

编译和解释方式是翻译高级程序设计语言的两种基本方式。

解释程序也称为解释器，它或者直接解释执行源程序，或者将源程序翻译成某种中间表示形式后再加以执行；而编译程序（编译器）则首先将源程序翻译成目标语言程序，然后在计算机上运行目标程序。这两种语言处理程序的根本区别是：在编译方式下，机器上运行的是与源程序等价的目标程序，源程序和编译程序都不再参与目标程序的执行过程；而在解释方式下，解释程序和源程序（或其某种等价表示）要参与到程序的运行过程中，运行程序的控制权在解释器。解释器翻译源程序时不产生独立的目标程序，而编译器则需将源程序翻译成独立的目标程序。

参考答案

（20）C

试题（21）

以下关于传值调用与引用调用的叙述中，正确的是__（21）__。

① 在传值调用方式下，可以实现形参和实参间双向传递数据的效果

② 在传值调用方式下，实参可以是变量，也可以是常量和表达式

③ 在引用调用方式下，可以实现形参和实参间双向传递数据的效果

④ 在引用调用方式下，实参可以是变量，也可以是常量和表达式

（21）A. ①③　　B. ①④　　C. ②③　　D. ②④

试题（21）分析

本题考查程序语言知识。

调用函数和被调用函数之间交换信息的方法主要有两种：一种是由被调用函数把返回值返回给主调函数，另一种是通过参数传递信息。函数调用时实参与形参间交换信息的基本方法有传值调用和引用调用两种。

若实现函数调用时实参向形参传递相应类型的值，则称为是传值调用。这种方式下形参不能向实参传递信息。实参可以是变量，也可以是常量和表达式。

引用调用的实质是将实参变量的地址传递给形参，因此，形参是指针类型，而实参必须具有左值。变量具有左值，常量没有左值。被调用函数对形参的访问和修改实际上就是针对相应实际参数所作的访问和改变，从而实现形参和实参间双向传递数据的效果。

参考答案

（21）C

试题（22）

在对高级语言源程序进行编译的过程中，为源程序中变量所分配的存储单元的地址属于<u>（22）</u>。

（22）A. 逻辑地址　　B. 物理地址　　C. 接口地址　　D. 线性地址

试题（22）分析

本题考查程序语言基础知识。

编译过程中为变量分配存储单元所用的地址是逻辑地址，程序运行时再映射为物理地址。

参考答案

（22）A

试题（23）

假设某分时系统采用简单时间片轮转法，当系统中的用户数为 n、时间片为 q 时，系统对每个用户的响应时间 T=<u>（23）</u>。

（23）A. n　　B. q　　C. n×q　　D. n+q

试题（23）分析

在分时系统中是将把 CPU 的时间分成很短的时间片轮流地分配给各个终端用户，当系统中的用户数为 n、时间片为 q 时，那么系统对每个用户的响应时间等于 n×q。

参考答案

（23）C

试题（24）

在支持多线程的操作系统中，假设进程 P 创建了若干个线程，那么＿（24）＿是不能被这些线程共享的。

（24）A．该进程的代码段　　B．该进程中打开的文件
　　　C．该进程的全局变量　　D．该进程中某线程的栈指针

试题（24）分析

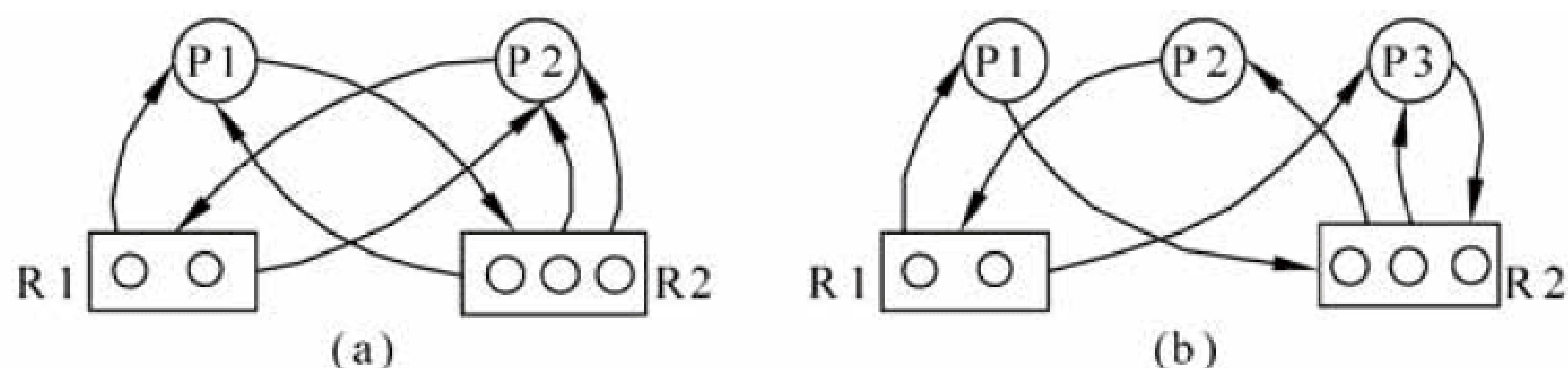
在同一进程中的各个线程都可以共享该进程所拥有的资源，如访问进程地址空间中的每一个虚地址；访问进程拥有已打开文件、定时器、信号量机构等，但是不能共享进程中某线程的栈指针。

参考答案

（24）D

试题（25）、（26）

进程资源图如图（a）和（b）所示，其中：图（a）中＿（25）＿；图（b）中＿（26）＿。



（25）A．P1 是非阻塞节点，P2 是阻塞节点，所以该图不可以化简、是死锁的
　　　B．P1、P2 都是阻塞节点，所以该图不可以化简、是死锁的
　　　C．P1、P2 都是非阻塞节点，所以该图可以化简、是非死锁的
　　　D．P1 是阻塞节点、P2 是非阻塞节点，所以该图不可以化简、是死锁的

（26）A．P1、P2、P3 都是非阻塞节点，该图可以化简、是非死锁的
　　　B．P1、P2、P3 都是阻塞节点，该图不可以化简、是死锁的
　　　C．P2 是阻塞节点，P1、P3 是非阻塞节点，该图可以化简、是非死锁的
　　　D．P1、P2 是非阻塞节点，P3 是阻塞节点，该图不可以化简、是死锁的

试题（25）、（26）分析

因为 R1 资源只有 2 个，P2 申请该资源得不到满足，故进程 P2 是阻塞节点；同样 R2 资源只有 3 个，P1 申请该资源得不到满足，故进程 P1 也是阻塞节点。可见进程资源图（a）是死锁的，该图不可以化简。

因为 R2 资源有 3 个，已分配 2 个，P3 申请 1 个 R2 资源可以得到满足，故进程 P3 可以运行完毕释放其占有的资源。这样可以使得 P1、P2 都变为非阻塞节点，得到所需资源运行完毕，因此，进程资源图（b）是可化简的。

参考答案

（25）B（26）C

试题（27）、（28）

假设内存管理采用可变式分区分配方案，系统中有五个进程 P1～P5，且某一时刻内存使用情况如下图所示（图中空白处表示未使用分区）。此时，若 P5 进程运行完并释放其占有的空间，则释放后系统的空闲区数应__（27）__；造成这种情况的原因是__（28）__。

分区号	进程
0	P1
1	P2
2	
3	P4
4	P3
5	
6	P5
7	

（27）A. 保持不变　　B. 减 1　　C. 加 1　　D. 置零

（28）A. 无上邻空闲区，也无下邻空闲区　　B. 有上邻空闲区，但无下邻空闲区

C. 有下邻空闲区，但无上邻空闲区　　D. 有上邻空闲区，也有下邻空闲区

试题（27）、（28）分析

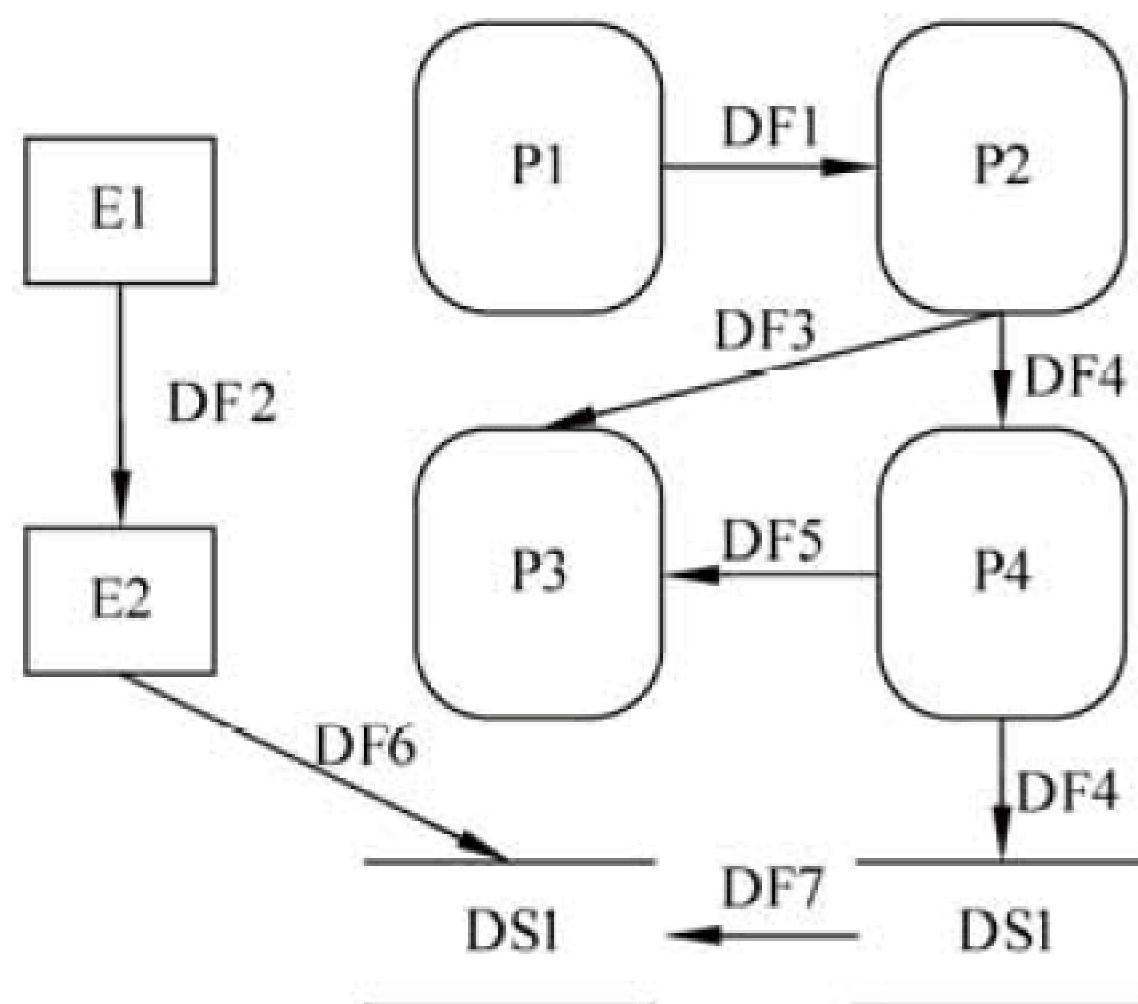
从图中不难看出，若 P5 进程运行完并释放其占有的空间，则由于其占用的分区有上邻空闲区，也有下邻空闲区，一旦释放后，就合并为一个空闲区，所以合并后系统空闲区数=3–1=2。

参考答案

（27）B（28）D

试题（29）

在如下所示的数据流图中，共存在__（29）__个错误。



（29）A．4　　B．6　　C．8　　D．9

试题（29）分析

本题考查数据流图方面的基础知识。

结构化分析将数据和处理作为分析对象，数据的分析结果表示了现实世界中实体的属性及其之间的相互关系，而处理的结果则展现了系统对数据的加工和转换。面向数据流建模是目前仍然被广泛使用的方法之一，而 DFD 则是面向数据流建模中的重要工具，DFD 将系统建模成输入—处理—输出的模型，即流入软件的数据对象，经由处理的转换，最后以结果数据对象的形式流出软件。在实际使用 DFD 进行数据流建模时，需要注意以下原则：

① 加工处理和数据流的正确使用，如一个加工必须既有输入又有输出；数据流只能和加工相关，即从加工流向加工、数据源流向加工或加工流向数据源。

② 每个数据流和数据存储都要在数据字典中有定义，数据字典将包括各层数据流图中数据元素的定义。

③ 数据流图中最底层的加工处理必须有加工处理说明。

④ 父图和子图必须平衡，即父图中某加工的输入输出（数据流）和分解这个加工的子图的输入输出数据流必须完全一致，这种一致性不一定要求数据流的名称和个数一一对应，但它们在数据字典中的定义必须一致，数据流或数据项既不能多也不能少。

⑤ 加工处理说明和数据流图中加工处理涉及的元素保持一致。例如，在加工处理说明中，输入数据流必须说明其如何使用，输出数据流说明如何产生或选取，数据存储说明如何选取、使用或修改。

⑥ 一幅图中的图元个数控制在 7+2 以内。

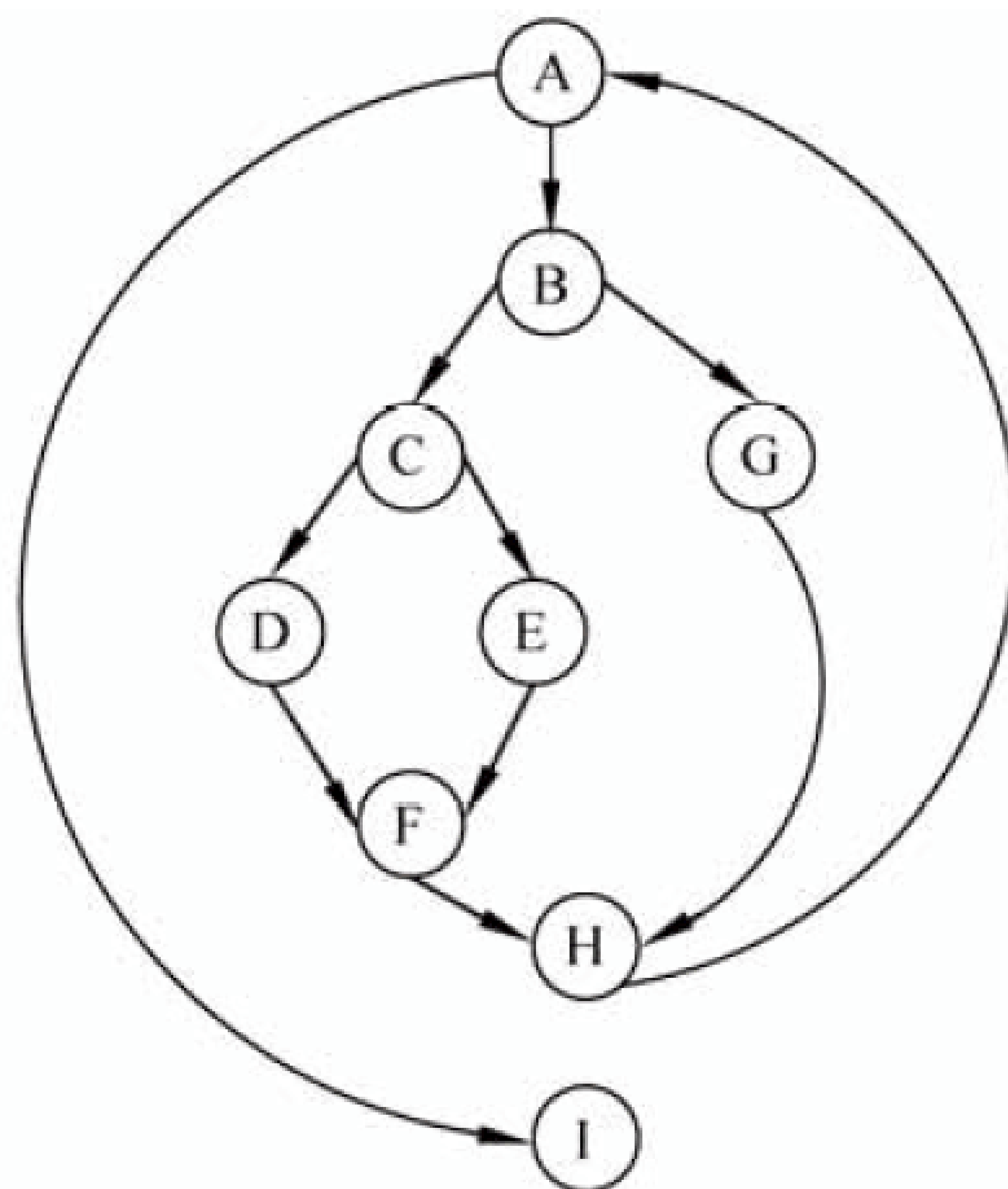
在题目所示的 DFD 图中，数据流 DF2、DF6 和 DF7 的输入、输出均不是加工，这与“数据流只能和加工相关，即从加工流向加工、数据源流向加工或加工流向数据源”相违背。加工 P1 只有输出，没有输入；加工 P3 只有输入没有输出，这与“一个加工必须既有输入又有输出”相违背。数据流 DF4 经过加工 P4 之后没有发生任何改变，说明该数据对加工 P4 是没有作用的，根据数据守恒原理，这条数据流不应与 P4 有关联。综上，该 DFD 图中共有 6 个错误。

参考答案

（29）B

试题（30）、（31）

软件的复杂性主要体现在程序的复杂性。__（30）__是度量软件复杂性的一个主要参数。若采用 McCabe 度量法计算环路复杂性，则对于下图所示的程序图，其环路复杂度为__（31）__。



（30）A. 代码行数　　　B. 常量的数量
　　　C. 变量的数量　　D. 调用的库函数的数量

（31）A. 2　　B. 3　　C. 4　　D. 5

试题（30）、（31）分析

本题考查软件复杂性方面的基础知识。

软件复杂性度量是软件度量的一个重要分支。对于软件复杂性度量的主要参数有：

- 规模，即总共的指令数，或源程序行数。
- 难度，通常由程序中出现的操作数的数目所决定的量来表示。
- 结构，通常用与程序结构有关的度量来表示。
- 智能度，即算法的难易程度。

软件复杂性主要表现在程序的复杂性。程序的复杂性主要指模块内程序的复杂性。McCabe 度量法是一种基于程序控制流的复杂性度量方法。McCabe 复杂性度量又称为环路度量，它认为程序的复杂性很大程度上取决于控制的复杂性。单一的顺序程序结构最为简单，循环和选择所构成的环路越多，程序就越复杂。这种方法以图论为工具，先画出程序图，然后用该图的环路数作为程序复杂性的度量值。程序图是退化的程序流程图，也就是说，把程序流程图中每个处理符号都退化成一个结点，原来连接不同处理符号的流线变成连接不同点的有向弧，这样得到的有向图就叫做程序图。程序图仅描述程序内部的控制流程，完全不表现对数据的具体操作以及分支和循环的具体条件。

根据图论，在一个强连通的有向图 G 中，环的个数 V(G)由以下公式给出：

$V(G) = m - n + 2p$

其中，V(G)是有向图 G 中的环路数，m 是图 G 中弧的个数，n 是图 G 中的结点数，p 是 G 中的强连通分量个数。在一个程序中，从程序图的入口点总能到达图中的任何一个结点，因此，程序总是连通的，但不是强连通的。为了使程序图成为强连通图，从图的入口点到出口点加一条用虚线表示的有向边，使图成为强连通图，这样就可以使用上

式计算环路复杂性了。对于题目中的程序图，其中结点数 n = 9，弧数 m = 11，p = 1，则有：

$$V(G) = m-n+2p = 11-9+2 = 4$$

即 McCabe 环路复杂的度量值为 4。

参考答案

（30）A　　（31）C

试题（32）

<u>（32）</u>不属于软件设计质量评审。

（32）A．功能与模块之间的对应关系　　B．软件规格说明是否符合用户的要求
　　　C．软件是否具有可测试性　　D．软件是否具有良好的可靠性

试题（32）分析

本题考查软件评审方面的基础知识。

通常，把“质量”理解为“用户满意程度”。为了使得用户满意，有两个必要条件：

① 设计的规格说明书符合用户的要求，这称为设计质量；

② 程序按照设计规格说明所规定的情况正确执行，这称为程序质量。

设计质量评审的对象是在需求分析阶段产生的软件需求规格说明、数据需求规格说明，在软件概要设计阶段产生的软件概要设计说明书等。通常从以下几个方面进行评审：

① 评价软件的规格说明是否合乎用户的要求，即总体设计思想和设计方针是否明确；需求规格说明是否得到了用户或单位上级机关的批准；需求规格说明与软件的概要设计规格说明是否一致等。

② 评审可靠性，即是否能避免输入异常（错误或超载等）、硬件失效及软件失效所产生的失效，一旦发生应能及时采取代替手段或恢复手段。

③ 评审保密措施实现情况，即是否对系统使用资格进行检查；是否对特定数据、特定功能的使用资格进行检查；在检查出有违反使用资格的情况后，能否向系统管理人员报告有关信息；是否提供对系统内重要数据加密的功能等。

④ 评审操作特性实施情况，即操作命令和操作信息的恰当性，输入数据与输入控制语句的恰当性；输出数据的恰当性；应答时间的恰当性等。

⑤ 评审性能实现情况，即是否达到所规定性能的目标值。

⑥ 评审软件是否具有可修改性，可扩充性、可互换性和可移植性。

⑦ 评审软件是否具有可测试性。

⑧ 评审软件是否具有复用性。

参考答案

（32）A

试题（33）

在软件维护中，由于企业的外部市场环境和管理需求的变化而导致的维护工作，属

于 (33) 维护。

(33) A. 正确性 B. 适应性 C. 完善性 D. 预防性

试题（33）分析

本题考查软件维护方面的基础知识。

软件维护主要是指根据需求变化或硬件环境的变化对应用程序进行部分或全部的修改。修改时应充分利用源程序，修改后要填写程序修改登记表，并在程序变更通知书上写明新老程序的不同之处。

软件维护的内容一般有以下几个方面：

① 正确性维护，是指改正在系统开发阶段已发生而系统测试阶段尚未发现的错误。这方面的维护工作量要占整个维护工作量的 17%～21%。所发现的错误有的不太重要，不影响系统的正常运行，其维护工作可随时进行；而有的错误非常重要，甚至影响整个系统的正常运行，其维护工作必须制定计划，进行修改，并且要进行复查和控制。

② 适应性维护，是指使应用软件适应信息技术变化和管理需求变化而进行的修改。这方面的维护工作量占整个维护工作量的 18%～25%。由于目前计算机硬件价格的不断下降，各类系统软件层出不穷，人们常常为改善系统硬件环境和运行环境而产生系统更新换代的需求；企业的外部市场环境和管理需求的不断变化也使得各级管理人员不断提出新的信息需求。这些因素都将导致适应性维护工作的产生。进行这方面的维护工作也要像系统开发一样，有计划、有步骤地进行。

③ 完善性维护，这是为扩充功能和改善性能而进行的修改，主要是指对已有的软件系统增加一些在系统分析和设计阶段中没有规定的功能与性能特征。这些功能对完善系统功能是非常必要的。另外还包括对处理效率和编写程序的改进，这方面的维护占整个维护工作的 50%～60%，比重较大，也是关系到系统开发质量的重要方面。这方面的维护除了要有计划、有步骤地完成外，还要注意将相关的文档资料加入到前面相应的文档中去。

④ 预防性维护，为了改进应用软件的可靠性和可维护性，以及适应未来的软硬件环境的变化，应主动增加预防性的新的功能，以使应用系统适应各类变化而不被淘汰。比如将专用报表功能改成通用报表生成功能，以适应将来报表格式的变化。这方面的维护工作量占整个维护工作量的 4%左右。

参考答案

(33) B

试题（34）

在对软件系统进行评价时，需要从信息系统的组成部分、评价对象和经济学角度出发进行综合考虑以建立起一套指标体系理论架构。从信息系统评价对象出发，对于用户方来说，他们所关心的是 (34) 。

(34) A. 用户需求和运行质量 B. 系统外部环境

C．系统内部结构　　　　　　　　D．系统质量和技术水平

试题（34）分析

本题考查系统评价方面的基础知识。

在对软件系统进行评价时，需要从信息系统的组成部分、评价对象和经济学角度出发进行综合考虑，以建立起一套指标体系理论架构。

从信息系统的组成部分出发，信息系统是一个由人机共同组成的系统，所以可以按照运行效果和用户需求（人）、系统质量和技术条件（机）这两条线索构造指标。

从信息系统评价对象出发，对于用户方来说，他们所关心的是用户需求和运行质量；对开发方而言，他们所关心的是系统质量和技术水平。系统外部环境则主要通过社会效益指标来反映。

从经济学角度出发，分别按系统成本、系统效益和财务指标等 3 条线索建立指标。

参考答案

（34）A

试题（35）

在设计测试用例时，应遵循＿（35）＿原则。

（35）A．仅确定测试用例的输入数据，无须考虑输出结果

B．只需检验程序是否执行了应有的功能，不需要考虑程序是否做了多余的功能

C．不仅要设计有效合理的输入，也要包含不合理、失效的输入

D．测试用例应设计得尽可能复杂

试题（35）分析

本题考查系统测试方面的基础知识。

系统测试是保证系统质量和可靠性的关键步骤，是对系统开发过程中的系统分析、系统设计和实施的最后复查。根据测试的概念和目的，在进行信息系统测试时应遵循以下基本原则：

① 应尽早并不断地进行测试。测试不是在应用系统开发完之后才进行的。由于原始问题的复杂性、开发各阶段的多样性以及参加人员之间的协调等因素，使得在开发各个阶段都有可能出现错误。因此，测试应贯穿在开发的各个阶段，尽早纠正错误，消除隐患。

② 测试工作应该避免由原开发软件的人或小组承担，一方面，开发人员往往不愿否认自己的工作，总认为自己开发的软件没有错误；另一方面，开发人员的错误很难由本人测试出来，很容易根据自己编程的思路来制定测试思路，具有局限性。测试工作应由专门人员来进行，会更客观，更有效。

③ 设计测试方案的时候，不仅要确定输入数据，而且要根据系统功能确定预期输出结果。将实际输出结果与预期结果相比较就能发现测试对象是否正确。

④ 在设计测试用例时，不仅要设计有效合理的输入条件，也要包含不合理、失效

的输入条件。测试的时候，人们往往习惯按照合理的、正常的情况进行测试，而忽略了对异常、不合理、意想不到的情况进行测试，而这些可能就是隐患。

⑤ 在测试程序时，不仅要检验程序是否做了该做的事，还要检验程序是否做了不该做的事。多余的工作会带来副作用，影响程序的效率，有时会带来潜在的危害或错误。

⑥ 严格按照测试计划来进行，避免测试的随意性。测试计划应包括测试内容、进度安排、人员安排、测试环境、测试工具和测试资料等。严格地按照测试计划可以保证进度，使各方面都得以协调进行。

⑦ 妥善保存测试计划、测试用例，作为软件文档的组成部分，为维护提供方便。

⑧ 测试用例都是精心设计出来的，可以为重新测试或追加测试提供方便。当纠正错误、系统功能扩充后，都需要重新开始测试，而这些工作重复性很高，可以利用以前的测试用例，或在其基础上修改，然后进行测试。

参考答案

（35）C

试题（36）

单元测试中，检查模块接口时，不需要考虑__（36）__。

（36）A. 测试模块的输入参数和形式参数在个数、属性、单位上是否一致

B. 全局变量在各模块中的定义和用法是否一致

C. 输入是否改变了形式参数

D. 输入参数是否使用了尚未赋值或者尚未初始化的变量

试题（36）分析

本题考查单元测试方面的基础知识。

单元测试也称为模块测试，在模块编写完成且无编译错误后就可以进行。单元测试侧重于模块中的内部处理逻辑和数据结构。单元测试主要检查模块的以下 5 个特征：模块接口、局部数据结构、重要的执行路径、出错处理和边界条件。

① 模块接口。模块的接口保证了测试模块的数据流可以正确地流入、流出。在测试中应检查以下要点：

- 测试模块的输入参数和形式参数在个数、属性、单位上是否一致。
- 调用其他模块时所给出的实际参数和被调用模块的形式参数在个数、属性、单位上是否一致。
- 调用标准函数时所用的参数在属性、数目和顺序上是否正确。
- 全局变量在各模块中的定义和用法是否一致。
- 输入是否仅改变了形式参数。
- 开/关的语句是否正确。
- 规定的 I/O 格式是否与输入输出语句一致。
- 在使用文件之前是否已经打开文件或是用文件之后是否已经关闭文件。

② 局部数据结构。在单元测试中，局部数据结构出错是比较常见的错误，在测试时应重点考虑以下因素：

- 变量的说明是否合适。
- 是否使用了尚未赋值或尚未初始化的变量。
- 变量的初始值或默认值是否正确。
- 变量名是否有错（例如：拼写错误）。

③ 重要的执行路径。在单元测试中，对路径的测试是最基本的任务。由于不能进行穷举测试，需要精心设计测试例子来发现是否有计算、比较或控制流等方面的错误。

- 计算方面的错误：算术运算的优先次序不正确或理解错误；精度不够；运算对象的类型彼此不相容；算法错误；表达式的符号表示不正确等。
- 比较和控制流的错误：本应相等的量由于精度造成不相等；不同类型进行比较；逻辑运算符不正确或优先次序错误；循环终止不正确（如多循环一次或少循环一次）、死循环；不恰当地修改循环变量；当遇到分支循环时，出口错误等。

④ 出错处理。好的设计应该能预测到出错的条件并且有对出错处理的路径。虽然计算机可以显示出错信息的内容，但仍需要程序员对出错进行处理，保证其逻辑的正确性，以便于用户维护。

⑤ 边界条件。边界条件的测试是单元测试的最后工作，也是非常重要的工作。软件容易在边界出现错误。

参考答案

（36）D

试题（37）

在多态的几种不同形式中，__（37）__多态是一种特定的多态，指同一个名字在不同上下文中可代表不同的含义。

（37）A.参数　　B. 包含　　C. 过载　　D. 强制

试题（37）分析

本题考查面向对象的基本知识。

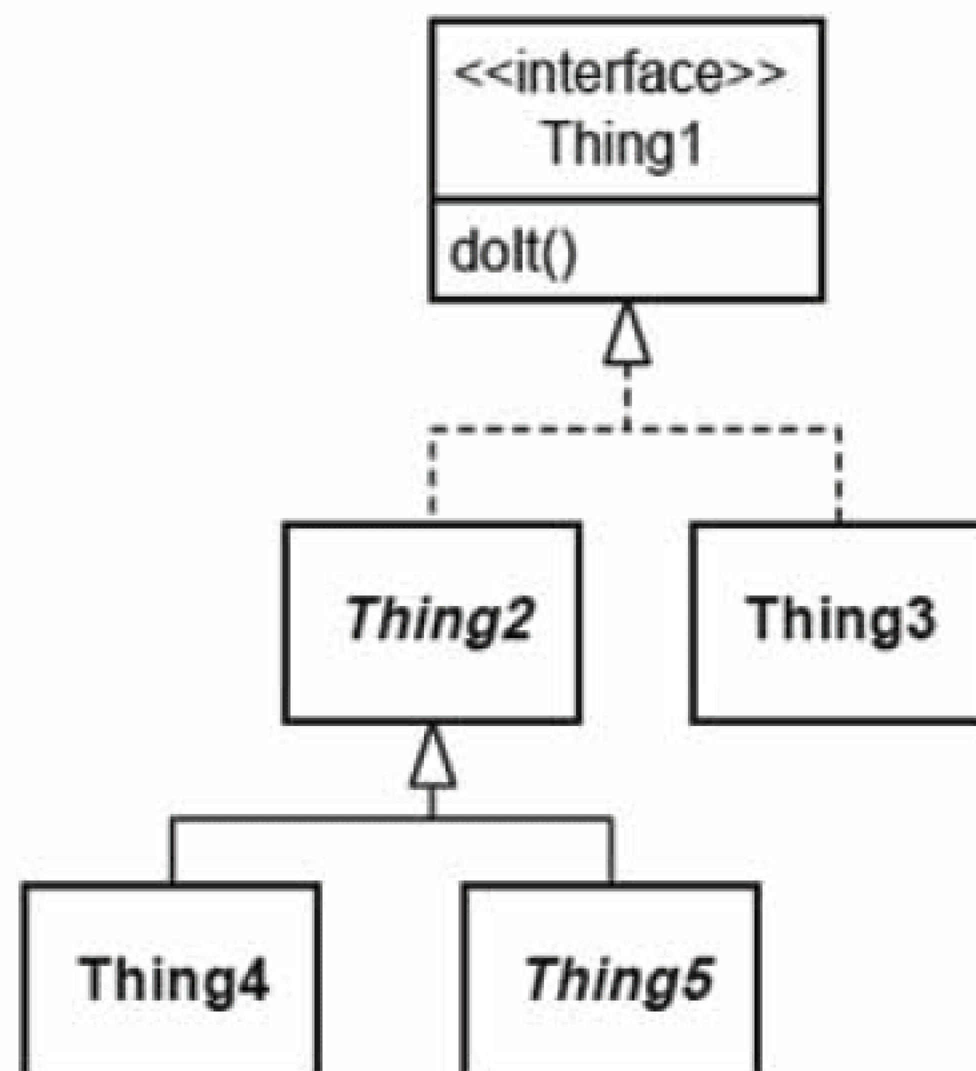
在面向对象技术中，不同的对象收到同一消息可以产生完全不同的结果，这一现象叫做多态（polymorphism）。在使用多态的时候，用户可以发送一个通用的消息，而实现的细节则由接收对象自行决定。这样，同一消息就可以调用不同的方法。多态有参数多态、包含多态、过载多态和强制多态四类。参数多态是应用比较广泛的多态，被称为最纯的多态，包含多态在许多语言中都存在，最常见的例子就是子类型化，即一个类型是另一个类型的子类型。过载多态是同一个名字在不同的上下文中所代表的含义不同。

参考答案

（37）C

试题（38）、（39）

继承是父类和子类之间共享数据和方法的机制。以下关于继承的叙述中，不正确的是__（38）__。有关下图中 doIt()方法的叙述中，正确的是__（39）__。



（38）A. 一个父类可以有多个子类，这些子类都是父类的特例
B. 父类描述了这些子类的公共属性和操作
C. 子类可以继承它的父类（或祖先类）中的属性和操作而不必自己定义
D. 子类中可以定义自己的新操作而不能定义和父类同名的操作

（39）A. doIt()必须由 Thing3 实现，同时可能由 Thing4 实现
B. doIt()必须由 Thing5 实现
C. doIt()必须由 Thing2、Thing3、Thing4 和 Thing5 实现
D. doIt()已经由 Thing1 实现，因此无须其他类实现

试题（38）、（39）分析

本题考查面向对象的基本知识。

面向对象技术中，继承关系是一种模仿现实世界中继承关系的一种类之间的关系，是超类（父类）和子类之间共享数据和方法的机制。父类定义公共的属性和操作，一个父类可以有多个子类，即多个特例。子类可以继承其父类或祖先类中的属性和操作作为自己的内容而不必自己定义，也可以覆盖这些操作，并加入新的内容。

接口是一种特殊的抽象机制，其中的操作不实现，需要由实现类来加以实现。对实现类为抽象类的，仍然可以保持操作为抽象，而如果是一个具体实现类，其中操作必须实现。题图中在接口 Thing1 中声明了 doIt()，在图中 Thing3 和 Thing2 作为 Thing1 的实现类，Thing3 为具体类，必须实现 doIt()；Thing2 可以保持 doIt()为抽象操作，由其子类实现此操作，Thing4 为具体类，可以实现 doIt()，而 Thing5 仍然是抽象类，也可以保持 doIt()为抽象操作。

参考答案

（38）D （39）A

试题（40）

以下关于 UML 部署图的叙述中，正确的是＿（40）＿。

（40）A．因为一条消息总是有某种响应，所以部署组件之间的依赖是双向的

B．部署组件之间的依赖关系类似于包依赖

C．部署图不用于描述代码的物理模块

D．部署图不用于描述系统在不同计算机系统的物理分布

试题（40）分析

本题考查统一建模语言（UML）的基本知识。

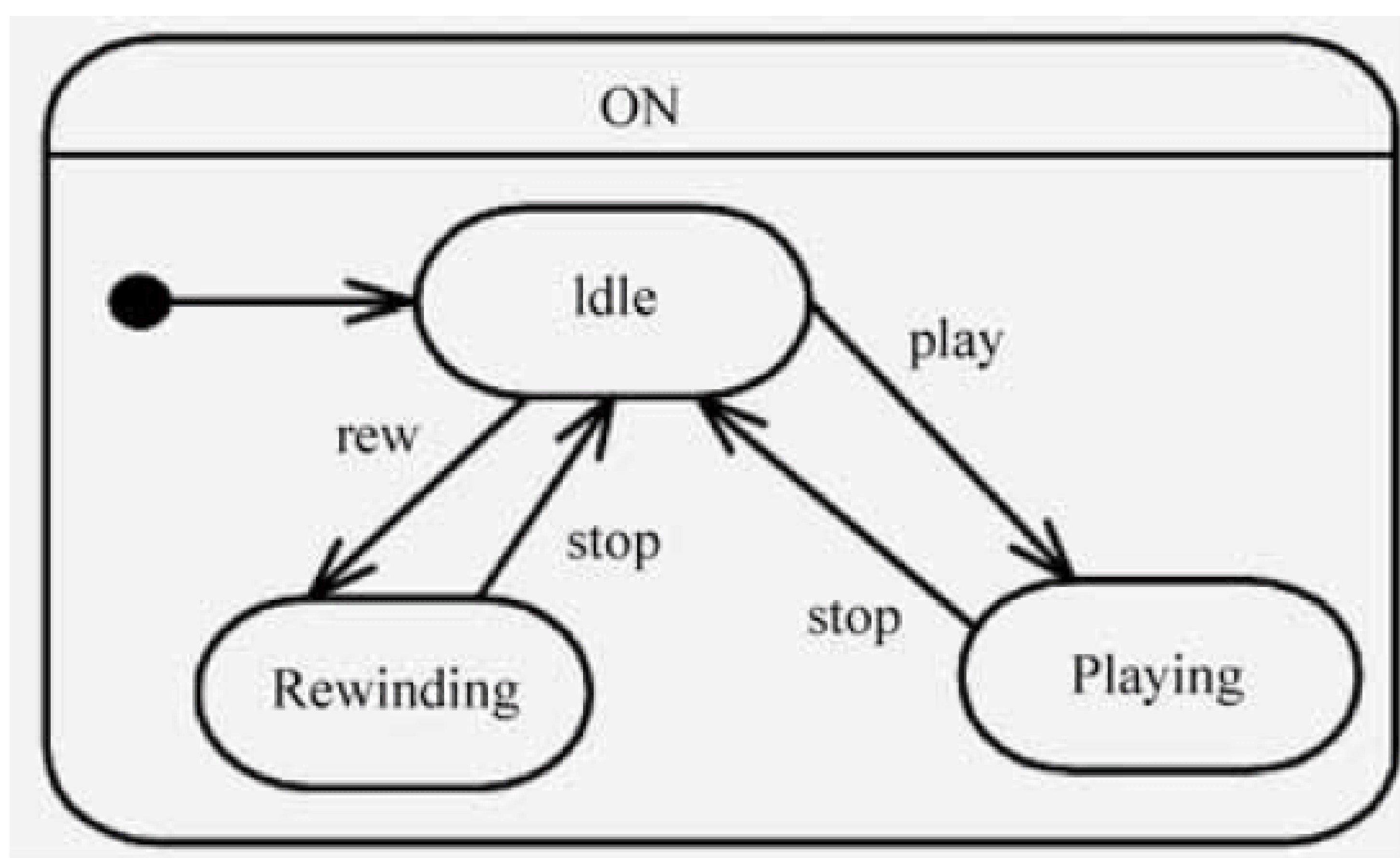
UML2.0 中提供的部署图展现了运行处理节点以及其中组件的配置，描述代码的物理模块，用于描述系统在不同计算机系统的物理分布。部署图给出了体系结构的静态实施视图。它与组件图相关，通常一个节点包含一个或多个组件，其依赖关系类似于包图。

参考答案

（40）B

试题（41）、（42）

以下关于 UML 状态图的叙述中，不正确的是＿（41）＿。对下图的描述正确的是＿（42）＿。



（41）A．用于描述一个对象在多个用例中的行为

B．用于某些具有多个状态的对象而不是系统中大多数或全部对象

C．用于描述多个对象之间的交互

D．可以用于用户界面或控制对象

（42）A．ON 是一个并发状态

B．因为此状态图中没有终止（final）状态，所以此图是无效的

C．play、stop 和 rew 是动作

D．ON 是超状态

试题（41）、（42）分析

本题考查采用统一建模语言（UML）进行面向对象建模的基本知识。

UML2.0 中状态图主要用于描述对象、子系统、系统的生命周期。通过状态图可以了解到一个对象所能到达的所有状态以及对象收到的事件（消息、超时、错误、条件满足等）对对象状态的影响等。针对具有可标记的状态和复杂的行为的对象构建状态图。状态可能有嵌套的子状态，且子状态可以是一个状态图。

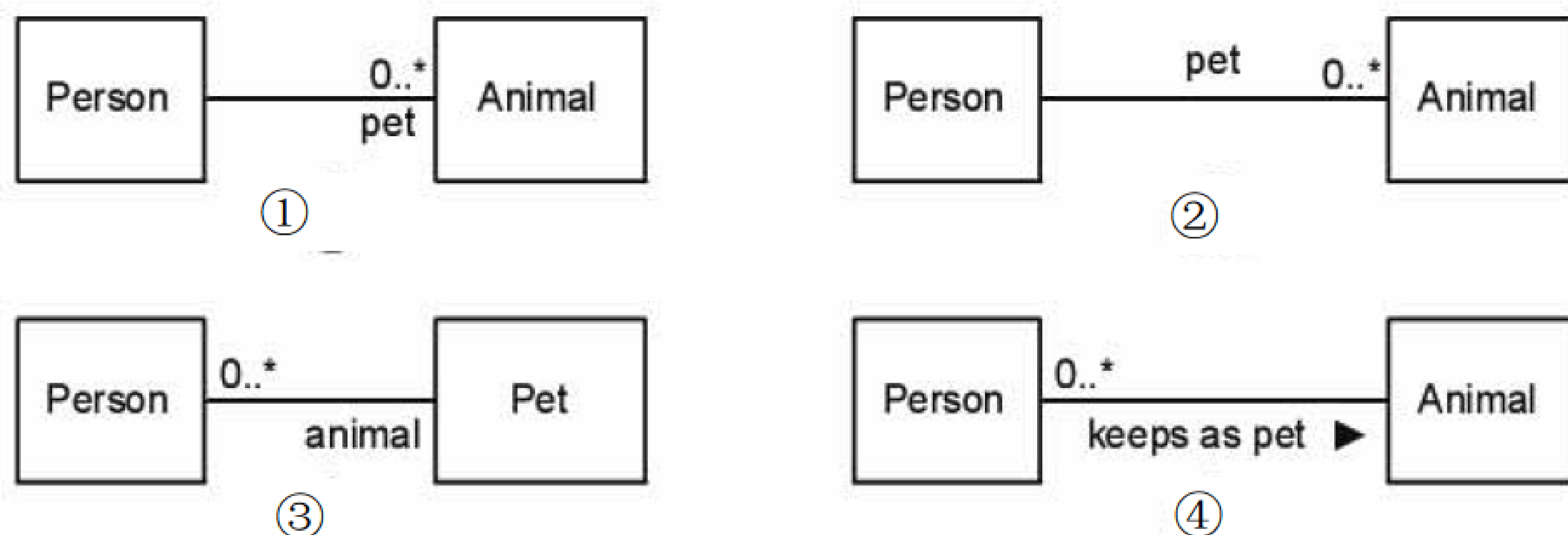
本题图示状态图中，ON 是一个超状态，它有 3 个子状态：Idle、Rewinding 和 Playing，这三个子状态之间在相关事件发生时状态之间进行迁移。

参考答案

（41）C　　（42）D

试题（43）

描述一些人（Person）将动物（Animal）养为宠物（Pet）的是图__（43）__。



（43）A. ①　　B. ②　　C. ③　　D. ④

试题（43）分析

本题考查统一建模语言（UML）的基本知识。

UML 中提供了类图，它以图形化的方式描述系统中对象及其关联关系。类图专注于系统的静态视图，它对于系统的领域内容建模特别重要。在该图中还包括多重性、关联关系、泛化/特化关系以及聚合关系。其中，关联关系是双向关系，即关联的对象双方彼此都能与对方通信。本题中人（Person）和动物（Animal）之间的关联关系，表示人可以将多只动物养为宠物（Pet），这时动物的角色是 pet，重数为 0..*。

参考答案

（43）A

试题（44）～（47）

__（44）__设计模式能使一个对象的状态发生改变时通知所有依赖它的监听者。__（45）__设计模式限制类的实例对象只能有一个。适配器（Adapter）设计模式可以用于__（46）__。用于为一个对象添加更多功能而不使用子类的是__（47）__设计模式。

（44）A. 责任链（Chain of Responsibility）　　B. 命令（Command）
　　C. 抽象工厂（Abstract Factory）　　D. 观察者（Observer）

（45）A. 原型（Prototype）　　B. 工厂方法（Factory Method）

C．单例（Singleton）　　D．生成器（Builder）

（46）A．将已有类的接口转换成和目标接口兼容

B．改进系统性能

C．将客户端代码数据转换成目标接口期望的合适的格式

D．使所有接口不兼容类可以一起工作

（47）A．桥接（Bridge）　　B．适配器（Adapter）

C．组合（Composite）　　D．装饰器（Decorator）

试题（44）～（47）分析

本题考查设计模式的基本概念。每种设计模式都有特定的意图，描述一个在我们周围不断重复发生的问题，以及该问题的解决方案的核心，使该方案能够重用而不必做重复劳动。

责任链（Chain of Responsibility）模式使多个对象都有机会处理请求，从而避免请求的发送者和接收者之间的耦合关系，将这些对象连成一条链，并沿着这条链传递该请求，直到有一个对象处理它为止。

命令（Command）模式将一个请求封装为一个对象，从而使得使用者可以采用不同的请求对客户进行参数化；对请求排队或记录请求日志，以及支持可撤销的操作。

抽象工厂（Abstract Factory）模式提供一个创建一系列相关或相互依赖对象的接口，而无须指定它们具体的类。

观察者（Observer）模式定义对象间的一种一对多的依赖关系，当一个对象的状态发生改变时，所有依赖于它的对象都得到通知并被自动更新。

原型（Prototype）模式用原型实例指定创建对象的种类，并且通过拷贝这个原型来创建新的对象。

工厂方法（Factory Method）定义一个用于创建对象的接口，让子类决定将哪一个类实例化，使一个类的实例化延迟到其子类。

单例（Singleton）模式是指系统运行过程中，一个类只有一个对象实例。

生成器（Builder）模式将一个复杂对象的构建与它的表示分离，使得同样的构建过程可以创建不同的表示。

适配器（Adapter）模式将一个类的接口转换成客户希望的另外一个接口，使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。

桥接（Bridge）模式将抽象部分与其实现部分分离，使它们都可以独立地变化。

组合（Composite）模式将对象组合成树形结构以表示“部分—整体”的层次结构，使得用户对单个对象和组合对象的使用具有一致性。

装饰器（Decorator）模式描述了以透明围栏来支持修饰的类和对象的关系，动态地给一个对象添加一些额外的职责，从增加功能的角度来看，装饰器模式相比生成子类更加灵活。

参考答案

（44）D （45）C （46）A （47）D

试题（48）

以下关于语言 $L=\{a^nb^n \mid n\geqslant 1\}$ 的叙述中，正确的是 （48） 。

（48）A. 可用正规式"aa^*bb^*"描述，但不能通过有限自动机识别

B. 可用正规式"a^mb^m"表示，也可用有限自动机识别

C. 不能用正规式表示，但可用有限自动机识别

D. 不能用正规式表示，也不能通过有限自动机识别

试题（48）分析

本题考查程序语言基础知识。

$L=\{a^nb^n \mid n\geqslant 1\}$ 中的字符串特点是 a 的个数与 b 的个数相同，且所有的 a 都在 b 之前，该集合不是正规集，不能用正规式表示。

正规集可用正规式描述，用有限自动机识别。

参考答案

（48）D

试题（49）、（50）

编译过程中，对高级语言程序语句的翻译主要考虑声明语句和可执行语句。对声明语句，主要是将所需要的信息正确地填入合理组织的 （49） 中；对可执行语句，则是 （50） 。

（49）A. 符号表 B. 栈 C. 队列 D. 树

（50）A. 翻译成机器代码并加以执行 B. 转换成语法树

C. 翻译成中间代码或目标代码 D. 转换成有限自动机

试题（49）、（50）分析

本题考查程序语言翻译基础知识。

符号表的作用是记录源程序中各个符号的必要信息，以辅助语义的正确性检查和代码生成，在编译过程中需要对符号表进行快速有效地查找、插入、修改和删除等操作。符号表的建立可以始于词法分析阶段，也可以放到语法分析和语义分析阶段，但符号表的使用有时会延续到目标代码的运行阶段。

编译过程中，在确认源程序的语法和语义之后，就可对其进行翻译，同时改变源程序的内部表示。对于声明语句，需要记录所遇到的符号的信息，因此应进行符号表的填查工作。对于可执行语句，需要翻译成中间代码或目标代码。

参考答案

（49）A （50）C

试题（51）

采用顺序表和单链表存储长度为 n 的线性序列，根据序号查找元素，其时间复杂度分别为__（51）__。

（51）A．O(1)、O(1)　B．O(1)、O(n)　C．O(n)、O(1)　D．O(n)、O(n)

试题（51）分析

本题考查数据结构基础知识。

对于长度为 n 的线性序列，若采用顺序表（一维数组）存储，则每个元素的位序与存储该元素的数组元素下标有直接的对应关系，可进行随机查找，时间复杂度为 O(1)；若采用单链表存储，则只能进行顺序访问，即必须从头指针出发，结合计数顺着指针链找到指定序号的元素，时间复杂度为 O(n)。

参考答案

（51）B

试题（52）

设元素序列 a、b、c、d、e、f 经过初始为空的栈 S 后，得到出栈序列 c e d f b a，则栈 S 的最小容量为__（52）__。

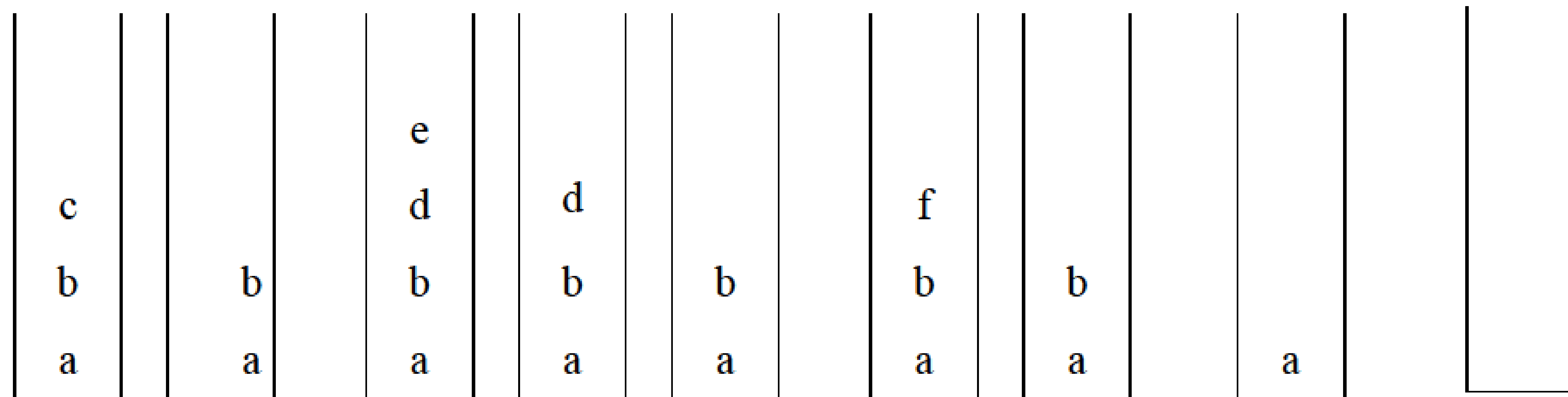
（52）A．3　B．4　C．5　D．6

试题（52）分析

本题考查数据结构基础知识。

栈是一种后进先出的数据结构。本问题中，根据元素入栈次序及出栈序列，每次需要出栈操作时栈的状态如下图所示，从中可以看出，栈中的元素个数最多时为 4。

出栈序列：c　　c e　c e d　　c e d f　c e d f b　c e d f b a

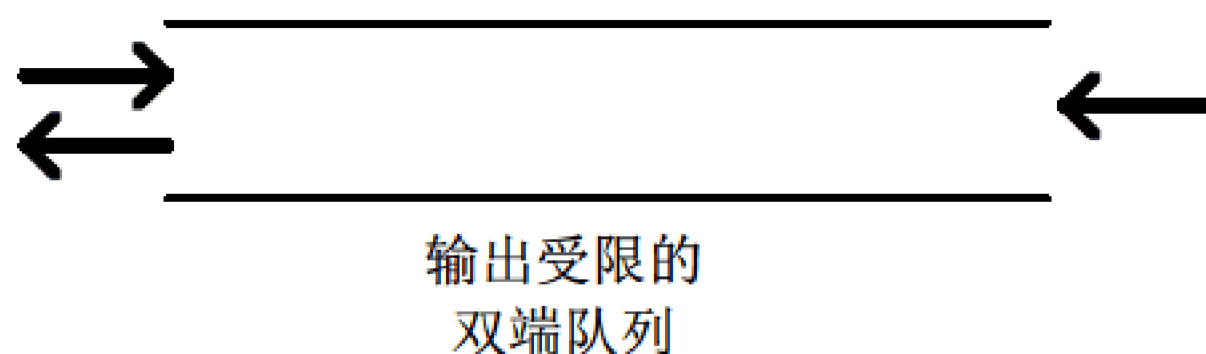


参考答案

（52）B

试题（53）

输出受限的双端队列是指元素可以从队列的两端输入、但只能从队列的一端输出，如下图所示。若有 e1、e2、e3、e4 依次进入输出受限的双端队列，则得不到输出序列__（53）__。



（53）A．e4、e3、e2、e1　　　　B．e4、e2、e1、e3

C．e4、e3、e1、e2　　　　D．e4、e2、e3、e1

试题（53）分析

本题考查数据结构基础知识。

该双端队列具有两个入口，所以 e1、e2、e3 进入队列后，从出口看可形成如下排列：

先出 e1，则得到 e1e2e3

先出 e2，则得到 e2e1e3

先出 e3，则得到 e3e1e2 或 e3e2e1

要在输出序列中首先得到 e4，元素 e4 只能从出口端进入队列，结合前三个元素的可能排列，因此以 e4 打头的输出序列有：e4e1e2e3，e4e2e1e3，e4e3e1e2，e4e3e2e1。

参考答案

（53）D

试题（54）

在数据库系统中，视图是一个__（54）__。

（54）A．真实存在的表，并保存了待查询的数据

B．真实存在的表，只有部分数据来源于基本表

C．虚拟表，查询时只能从一个基本表中导出

D．虚拟表，查询时可以从一个或者多个基本表或视图中导出

试题（54）分析

本题考查数据库系统概念方面的基本概念。

在数据库系统中，当视图创建完毕后，数据字典中存放的是视图定义。视图是从一个或者多个表或视图中导出的表，其结构和数据是建立在对应表的查询基础上的。和真实的表一样，视图也包括几个被定义的数据列和多个数据行，但从本质上讲，这些数据列和数据行来源于其所引用的表。因此，视图不是真实存在的基础表，而是一个虚拟表，视图所对应的数据并不实际地以视图结构存储在数据库中，而是存储在视图所引用的基本表中。

参考答案

（54）D

试题（55）

给定关系模式 R(U，F)，其中：属性集 U＝{A,B,C,D,E,G}，函数依赖集 F={ A→B，A→C，C→D，AE→G}。若将 R 分解为如下两个子模式__（55）__，则分解后的关系模

式保持函数依赖。

（55）A．R1(A,B,C) 和 R2(D,E,G)　　B．R1(A,B,C,D) 和 R2(A,E,G)

C．R1(B,C,D) 和 R2(A,E,G)　　D．R1(B,C,D,E) 和 R2(A,E,G)

试题（55）分析

本题考查关系模式和关系规范化方面的基础知识。

根据题意可以求出 R1(A,B,C,D)的函数依赖集 F1={A→B，A→C，C→D}，R2(A,E,G)的函数依赖集 F2={AE→G}，而 F= F1+ F2，所以分解后的关系模式保持函数依赖。

参考答案

（55）B

试题（56）～（59）

假设学生 Students 和教师 Teachers 的关系模式如下所示：

Students(学号，姓名，性别，类别，身份证号)

Teachers(教师号，姓名，性别，身份证号，工资)

其中，学生关系中的类别分为“本科生”和“研究生”两类。

a．查询在读研究生的教师的平均工资、最高与最低工资之间差值的 SQL 语句如下：

SELECT ＿＿＿（56）＿＿＿

FROM　Students，Teachers

WHERE ＿＿＿（57）＿＿＿；

（56）A．AVG（工资）AS 平均工资，MAX（工资）－MIN（工资）AS 差值

B．平均工资 AS AVG（工资），差值 AS MAX（工资）－MIN（工资）

C．AVG（工资）ANY 平均工资，MAX（工资）－MIN（工资）ANY 差值

D．平均工资 ANY AVG（工资），差值 ANY MAX（工资）－MIN（工资）

（57）A．Students.身份证号= Teachers.身份证号

B．Students.类别='研究生'

C．Students.身份证号= Teachers.身份证号 AND Students.类别='研究生'

D．Students.身份证号= Teachers.身份证号 OR Students.类别='研究生'

b．查询既是女教师，又是研究生且工资大于等于 3500 元的身份证号和姓名的 SQL 语句如下：

(SELECT　身份证号，姓名

FROM　Students

WHERE ＿＿＿（58）＿＿＿)

INTERSECT

(SELECT　身份证号，姓名

FROM　Teachers

WHERE ＿＿＿（59）＿＿＿)

（58）A．工资>=3500　　B．工资>='3500'
C．性别=女 AND 类别=研究生　　D．性别='女' AND 类别='研究生'
（59）A．工资 >=3500　　B．工资 >='3500'
C．性别=女 AND 类别=研究生　　D．性别='女' AND 类别='研究生'

试题（56）～（59）分析

本题考查 SQL 方面的基础知识。

SQL 提供可为关系和属性重新命名的机制，这是通过使用具有“Old-name as new-name”形式的 as 子句来实现的。As 子句既可出现在 select 子句，也可出现在 from 子句中。

查询在读研究生的教师的平均工资、最高与最低工资之间差值需要用条件“Students.院系= Teachers.院系 AND Students.类别='研究生'”来限定。

因为第一条 SELECT 语句是从 Students 关系中查找女研究生的姓名和通信地址，故用条件“性别='女' AND 类别='研究生'”来限定；第二条 SELECT 语句查询是从 Teachers 关系中查找工资大于等于 3500 元的教师的姓名和通信地址，故用条件“工资 >=3500”限定。又因为，第一条 SELECT 语句查询和第二条 SELECT 语句查询的结果集模式都为（姓名，通信地址），故可以用“INTERSECT”对它们取交集。

参考答案

（56）A　（57）C　（58）D　（59）A

试题（60）、（61）

考虑下述背包问题的实例。有 5 件物品，背包容量为 100，每件物品的价值和重量如下表所示，并已经按照物品的单位重量价值从大到小排好序。根据物品单位重量价值大优先的策略装入背包中，则采用了__（60）__设计策略。考虑 0/1 背包问题（每件物品或者全部装入背包或者不装入背包）和部分背包问题（物品可以部分装入背包），求解该实例得到的最大价值分别为__（61）__。

物品编号	价值	重量
1	50	5
2	200	25
3	180	30
4	225	45
5	200	50

（60）A．分治　　B．贪心　　C．动态规划　　D．回溯
（61）A．605 和 630　　B．605 和 605　　C．430 和 630　　D．630 和 430

试题（60）、（61）分析

本题考查算法设计和分析知识。

背包问题是典型的算法问题，包括两种形式，即 0-1 背包问题和部分背包问题。0-1 背包问题是指每个物品或者全部放在背包中或者不放在背包中，求解在特定背包容量下装入背包物品的最大价值。部分背包问题中，每个物品可以部分地放入背包中，求解在特定背包容量下装入背包物品的最大价值。

基于单位重量价值最大优先的策略来将物品放入背包中，本质上是一种贪心的策略。在该策略下求 0-1 背包问题，不能确保得到最优解，事实上在本题给出实例中是得不到最优解的。而对于部分背包问题，是可以得到最优解的。

基于单位重量价值最大优先策略求解本题给出的实例。对于 0-1 背包问题，首先将物品 1、2 和 3 放入背包中，4 和 5 都不能再放入背包，此时背包重量为 5+25+30=60，获得价值 50+200+180=430。对于部分背包问题首先将物品 1、2 和 3 放入背包中，此时背包重量为 60，获得价值 430，此时还有剩余容量 100−60=40，可以将部分物品 4 放入背包，放入 40/45=8/9 的物品 4，价值为 225*8/9=200，因此得到的总价值为 430+200=630。

参考答案

（60）B　（61）C

试题（62）、（63）

给定 n 个整数构成的数组 $A=\{a_1,a_2,\ldots,a_n\}$和整数 x，判断 A 中是否存在两个元素 a_i 和 a_j，使得 $a_i+a_j=x$。为了求解该问题，首先用归并排序算法对数组 A 进行从小到大排序；然后判断是否存在 $a_i+a_j=x$，具体的方法如下列伪代码所示。则求解该问题时排序算法应用了__（62）__算法设计策略，整个算法的时间复杂度为__（63）__。

```
…
i = 1; j = n
while i < j
    if ai+aj = x return true
    else if ai+aj > x
        j--;
    else
        i++;
return false;
```

（62）A．分治　　B．贪心　　C．动态规划　　D．回溯

（63）A．$O(n)$　　B．$O(n\lg n)$　　C．$O(n^2)$　　D．$O(n\lg n^2)$

试题（60）、（61）分析

本题考查算法设计和分析知识。

本题给出的问题求解算法包括两个部分，归并排序和搜索元素。归并排序是一个采用分治策略的经典排序算法；而搜索过程则是从两端往里判断是否存在 $a_i+a_j=x$，此过程不涉及分治、贪心、动态规划和回溯等策略。因此算法采用的是分治策略。

算法的时间复杂度也是从两个部分分析得到的。归并排序的时间复杂度为 O(nlgn)；而搜索过程的时间复杂度为 O(n)。因此算法的时间复杂度为 O(nlgn)。

参考答案

（62）A （63）B

试题（64）

一个高度为 h 的满二叉树的结点总数为 2^h-1，从根结点开始，自上而下、同层次结点从左至右，对结点按照顺序依次编号，即根结点编号为 1，其左、右孩子结点编号分别为 2 和 3，再下一层从左到右的编号为 4、5、6、7，依此类推。那么，在一棵满二叉树中，对于编号为 m 和 n 的两个结点，若 n=2m+1，则__（64）__结点。

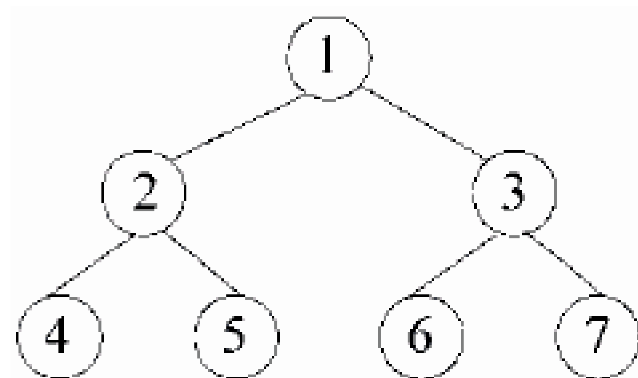
（64）A．m 是 n 的左孩子 B．m 是 n 的右孩子

C．n 是 m 的左孩子 D．n 是 m 的右孩子

试题（64）分析

本题考查数据结构基础知识。

用验证的方法求解，以高度为 3 的满二叉树（如下图所示）为例进行说明。



从中可以看出，若 n=2m+1，则结点 n 是 m 的右孩子结点。

参考答案

（64）D

试题（65）

以下关于哈希（Hash，散列）查找的叙述中，正确的是__（65）__。

（65）A．哈希函数应尽可能复杂些，以消除冲突

B．构造哈希函数时应尽量使关键字的所有组成部分都能起作用

C．进行哈希查找时，不再需要与查找表中的元素进行比较

D．在哈希表中只能添加元素不能删除元素

试题（65）分析

本题考查数据结构基础知识。

哈希表是通过一个以记录的关键字为自变量的函数（称为哈希函数）得到该记录的存储地址而构造的查找表，所以在哈希表中进行查找操作时，必须用同一哈希函数计算得到待查记录的存储地址，然后到相应的存储单元里去获得有关信息再判定查找是否成功。

冲突是指哈希函数将关键字不同的元素映射到了同一个存储地址。要减少冲突，就要设法使哈希函数尽可能均匀地把关键字映射到存储区的各个存储地址上，这样就可以

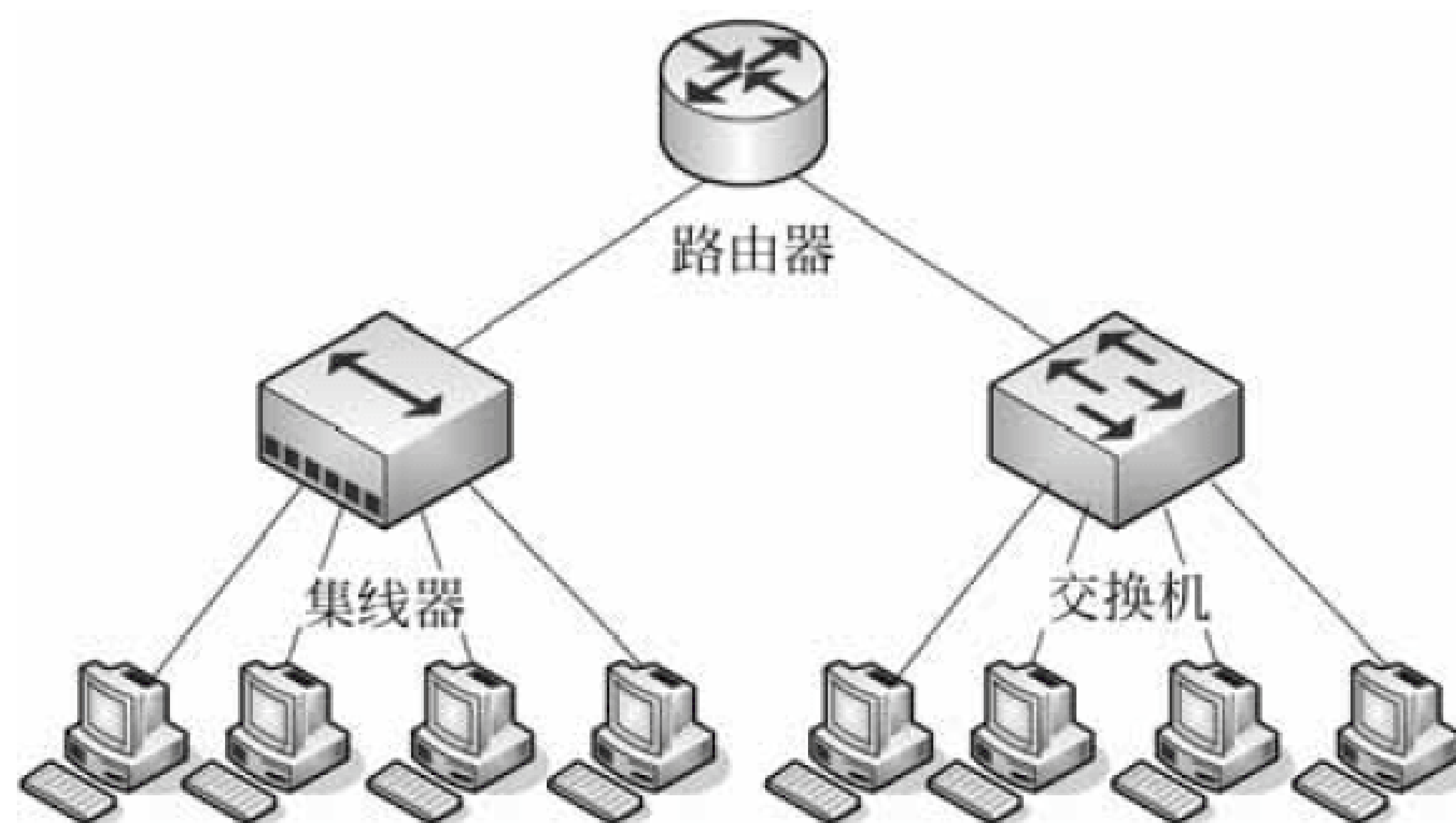
提高查找效率。构造哈希函数时，一般应使关键字的所有组成部分都能起作用。

参考答案

（65）B

试题（66）

网络配置如下图所示，其中使用了一台路由器、一台交换机和一台集线器，对于这种配置，下面的论断中正确的是＿（66）＿。



（66）A．2 个广播域和 2 个冲突域　　B．1 个广播域和 2 个冲突域
C．2 个广播域和 5 个冲突域　　D．1 个广播域和 8 个冲突域

试题（66）分析

集线器连接的主机构成一个冲突域，交换机的每个端口属于一个冲突域，路由器连接的两部分网络形成两个广播域，所以共有两个广播域和 5 个冲突域。

参考答案

（66）C

试题（67）、（68）

把网络 117.15.32.0/23 划分为 117.15.32.0/27，则得到的子网是＿（67）＿个。每个子网中可使用的主机地址是＿（68）＿个。

（67）A．4　　B．8　　C．16　　D．32

（68）A．30　　B．31　　C．32　　D．34

试题（67）、（68）分析

把网络 117.15.32.0/23 划分为 117.15.32.0/27，则子网掩码扩大了 4 位，所以得到的子网是 16 个。由于子网掩码为 27 位，所以主机地址只占 5 位，每个子网中可使用的主机地址是 30 个。

参考答案

（67）C　（68）A

试题（69）

通常工作在 UDP 协议之上的应用是＿（69）＿。

（69）A．浏览网页　　B．Telnet 远程登录　　C．VoIP　　D．发送邮件

试题（69）分析

本试题考查各网络应用采用的下层传输协议。

浏览网页、Telnet 远程登录以及发送邮件应用均不允许数据的丢失，需要采用可靠的传输层协议 TCP，而 VoIP 允许某种程度上的数据丢失，采用不可靠的传输层协议 UDP。

参考答案

（69）C

试题（70）

随着网站知名度不断提高，网站访问量逐渐上升，网站负荷越来越重，针对此问题，一方面可通过升级网站服务器的软硬件，另一方面可以通过集群技术，如 DNS 负载均衡技术来解决。在 Windows 的 DNS 服务器中通过 __(70)__ 操作可以确保域名解析并实现负载均衡。

（70）A．启用循环，启动转发器指向每个 Web 服务器
B．禁止循环，启动转发器指向每个 Web 服务器
C．禁止循环，添加每个 Web 服务器的主机记录
D．启用循环，添加每个 Web 服务器的主机记录

试题（70）分析

本题考查 Windows 的 DNS 服务器实现负载均衡的相关操作。

在 Windows 的 DNS 服务器中基于 DNS 的循环（round robin），只需要为同一个域名设置多个 ip 主机记录就可以了，DNS 中没有转发器的概念，因此需要启用循环，添加每个 Web 服务器的主机记录就可以确保域名解析并实现负载均衡。

参考答案

（70）D

试题（71）～（75）

So it is today. Schedule disaster, functional misfits, and system bugs all arise because the left hand doesn't know what the right hand is doing. As work __(71)__, the several teams slowly change the functions, sizes, and speeds of their own programs, and they explicitly or implicitly __(72)__ their assumptions about the inputs available and the uses to be made of the outputs.

For example, the implementer of a program-overlaying function may run into problems and reduce speed relying on statistics that show how __(73)__ this function will arise in application programs. Meanwhile, back at the ranch, his neighbor may be designing a major part of the supervisor so that it critically depends upon the speed of this function. This change in speed itself becomes a major specification change, and it needs to be proclaimed abroad and weighed from a system point of view.

How, then, shall teams __(74)__ with one another? In as many ways as possible.

- Informally. Good telephone service and a clear definition of intergroup dependencies will encourage the hundreds of calls upon which common interpretation of written documents depends.
- Meetings. Regular project meetings, with one team after another giving technical briefings, are ___(75)___. Hundreds of minor misunderstandings get smoked out this way.
- Workbook. A formal project workbook must be started at the beginning.

（71）A. starts B. proceeds C. stops D. speeds
（72）A. change B. proceed C. smooth D. hide
（73）A. frequently B. usually C. commonly D. rarely
（74）A. work B. program C. communicate D. talk
（75）A. worthless B. valueless C. useless D. invaluable

参考译文

现在，其实也是这样的情况。因为左手不知道右手在做什么，所以进度灾难、功能的不合理和系统缺陷纷纷出现。随着工作的进行，许多小组慢慢地修改自己程序的功能、规模和速度，他们明确或者隐含地更改了一些有效输入和输出结果用法上的约定。

例如，程序覆盖（program-overlay）功能的实现者遇到了问题，并且统计报告显示了应用程序很少使用该功能。基于这些考虑，他降低了覆盖功能的速度。与此同时，整个开发队伍中，其他同事正在设计监控程序。监控程序在很大程度上依赖于覆盖功能，它在速度上的变化成为了主要的规格说明变更。因此需要从系统角度来考虑和衡量该变化，以及公开、广泛地发布变更结果。

那么，团队如何进行相互之间的交流沟通呢？通过所有可能的途径。

非正式途径，清晰定义小组内部的相互关系和充分利用电话，能鼓励大量的电话沟通，从而达到对所书写文档的共同理解。

会议，常规项目会议。会议中，团队一个接一个地进行简要的技术陈述。这种方式非常有用，能澄清成百上千的细小误解。

工作手册，在项目的开始阶段，应该准备正式的项目工作手册。

参考答案

（71）B （72）A （73）D （74）C （75）D

第 18 章　2013 上半年软件设计师下午试题分析与解答

试题一（共 15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某慈善机构欲开发一个募捐系统，以跟踪记录为事业或项目向目标群体进行募捐而组织的集体性活动。该系统的主要功能如下所述。

（1）管理志愿者。根据募捐任务给志愿者发送加入邀请、邀请跟进、工作任务；管理志愿者提供的邀请响应、志愿者信息、工作时长、工作结果等。

（2）确定募捐需求和收集所募捐赠（资金及物品）。根据需求提出募捐任务、活动请求和捐赠请求，获取所募集的资金和物品。

（3）组织募捐活动。根据活动请求，确定活动时间范围。根据活动时间，搜索场馆，即：向场馆发送场馆可用性请求，获得场馆可用性。然后根据活动时间和地点推广募捐活动，根据相应的活动信息举办活动，从募款机构获取资金并向其发放赠品。获取和处理捐赠，根据捐赠请求，提供所募集的捐赠；处理与捐赠人之间的交互，即：

录入捐赠人信息，处理后存入捐赠人信息表；从捐赠人信息表中查询捐赠人信息，向捐赠人发送募捐请求，并将已联系的捐赠人存入已联系的捐赠人表。根据捐赠请求进行募集，募得捐赠后，将捐赠记录存入捐赠表；对捐赠记录进行处理后，存入已处理捐赠表，向捐赠人发送致谢函。根据已联系的捐赠人和捐赠记录进行跟进，将捐赠跟进情况发送给捐赠人。

现采用结构化方法对募捐系统进行分析与设计，获得如图 1-1、1-2 和 1-3 所示分层数据流图。

【问题 1】（4 分）

使用说明中的词语，给出图 1-1 中的实体 E1～E4 的名称。

【问题 2】（7 分）

在建模 DFD 时，需要对有些复杂加工（处理）进行进一步精化，图 1-2 为图 1-1 中处理 3 的进一步细化的 1 层数据流图，图 1-3 为图 1-2 中 3.1 进一步细化的 2 层数据流图。补全图 1-2 中加工 P1、P2 和 P3 的名称和图 1-2 与图 1-3 中缺少的数据流。

【问题 3】（4 分）

使用说明中的词语，给出图 1-3 中的数据存储 D1～D4 的名称。

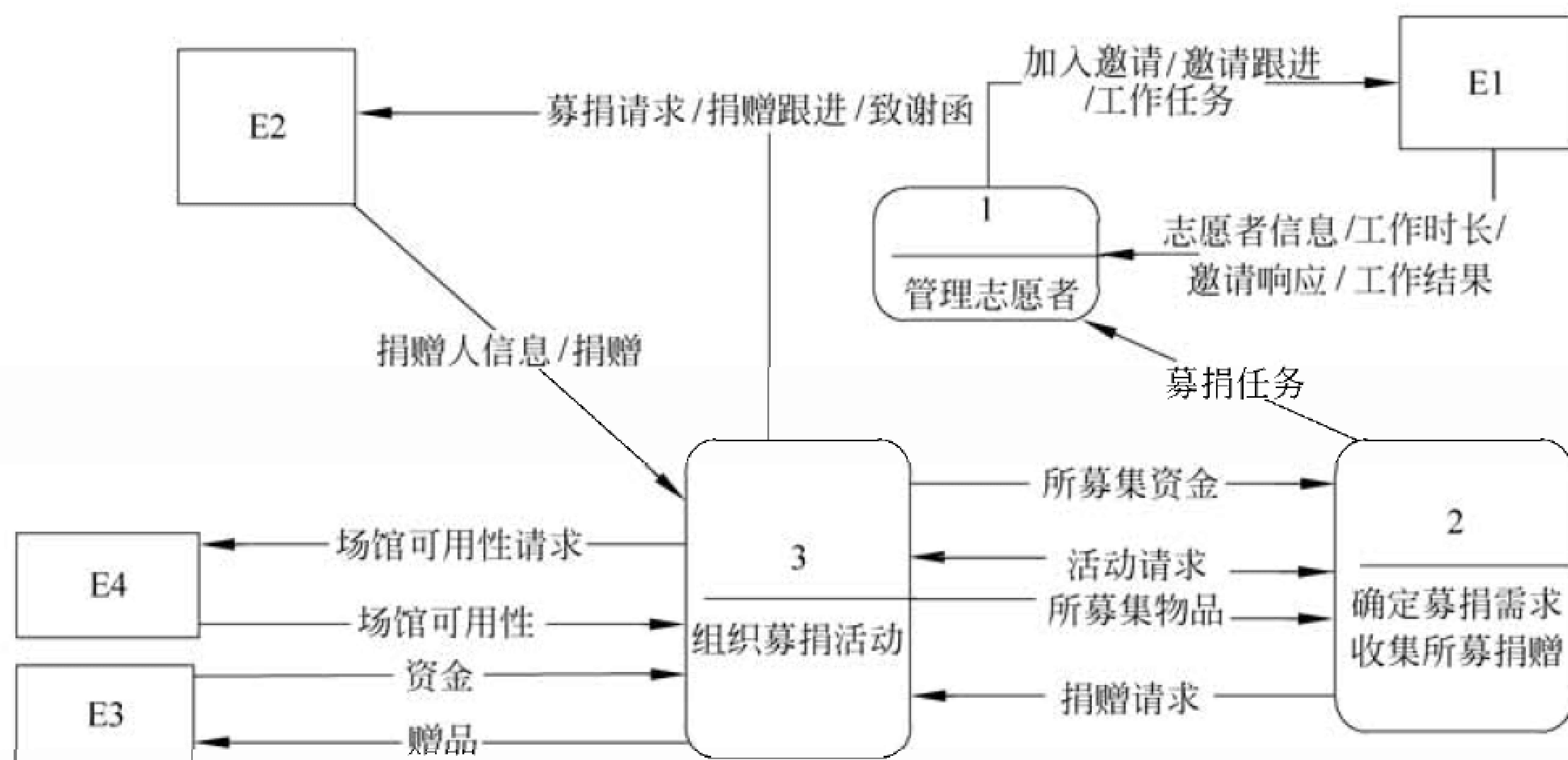


图 1-1　0 层数据流图

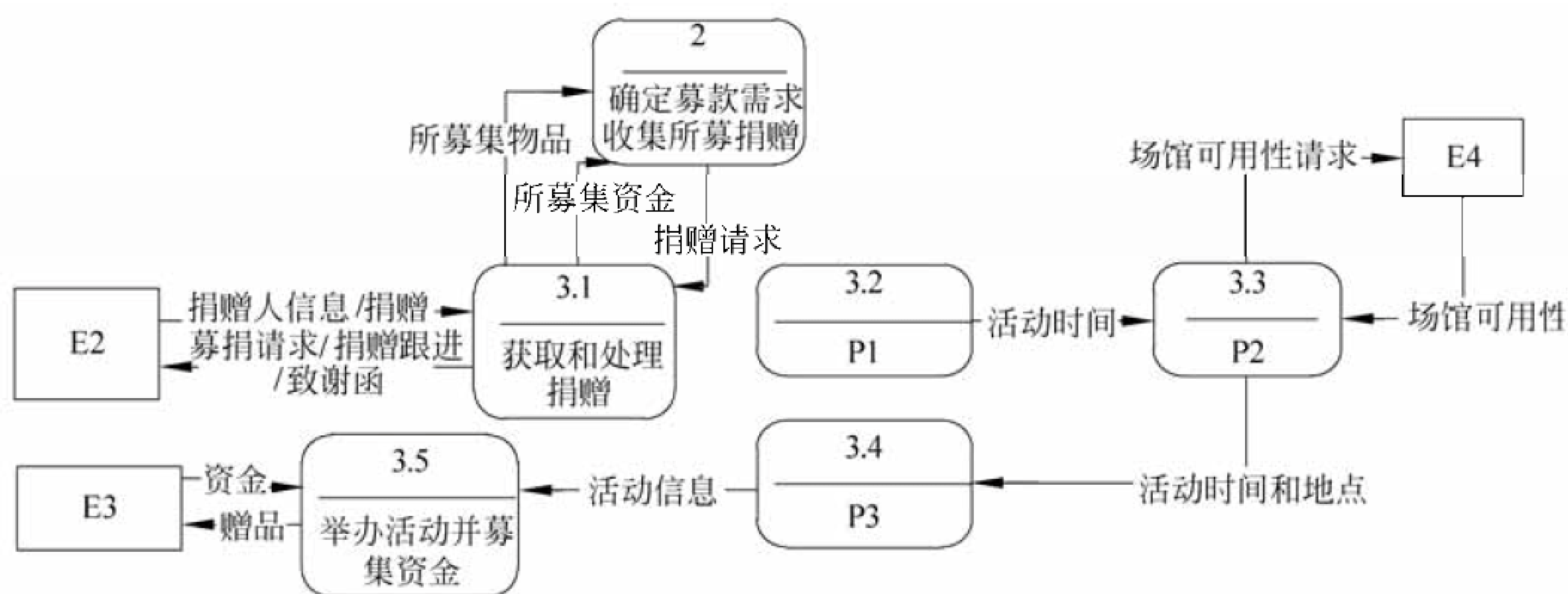


图 1-2　1 层数据流图

试题一分析

本题采用结构化方法进行系统分析与设计，主要考查数据流图（DFD）的应用，是比较传统的题目，要求考生细心分析题目中所描述的内容。

DFD 是一种便于用户理解、分析系统数据流程的图形化建模工具，是系统逻辑模型的重要组成部分。顶层 DFD 一般用来确定系统边界，将待开发系统看作一个大的加工（处理），然后根据系统从哪些外部实体接收数据流，以及系统将数据流发送到哪些外部实体，建模出的顶层图中只有唯一的一个加工和一些外部实体，以及这两者之间的输入输出数据流。0 层 DFD 在顶层确定的系统外部实体以及与外部实体的输入输出数据流的基础上，将顶层 DFD 中的加工分解成多个加工，识别这些加工的输入输出数据流，使

得所有顶层 DFD 中的输入数据流，经过这些加工之后变换成顶层 DFD 的输出数据流。根据 0 层 DFD 中的加工的复杂程度进一步建模加工的内容。

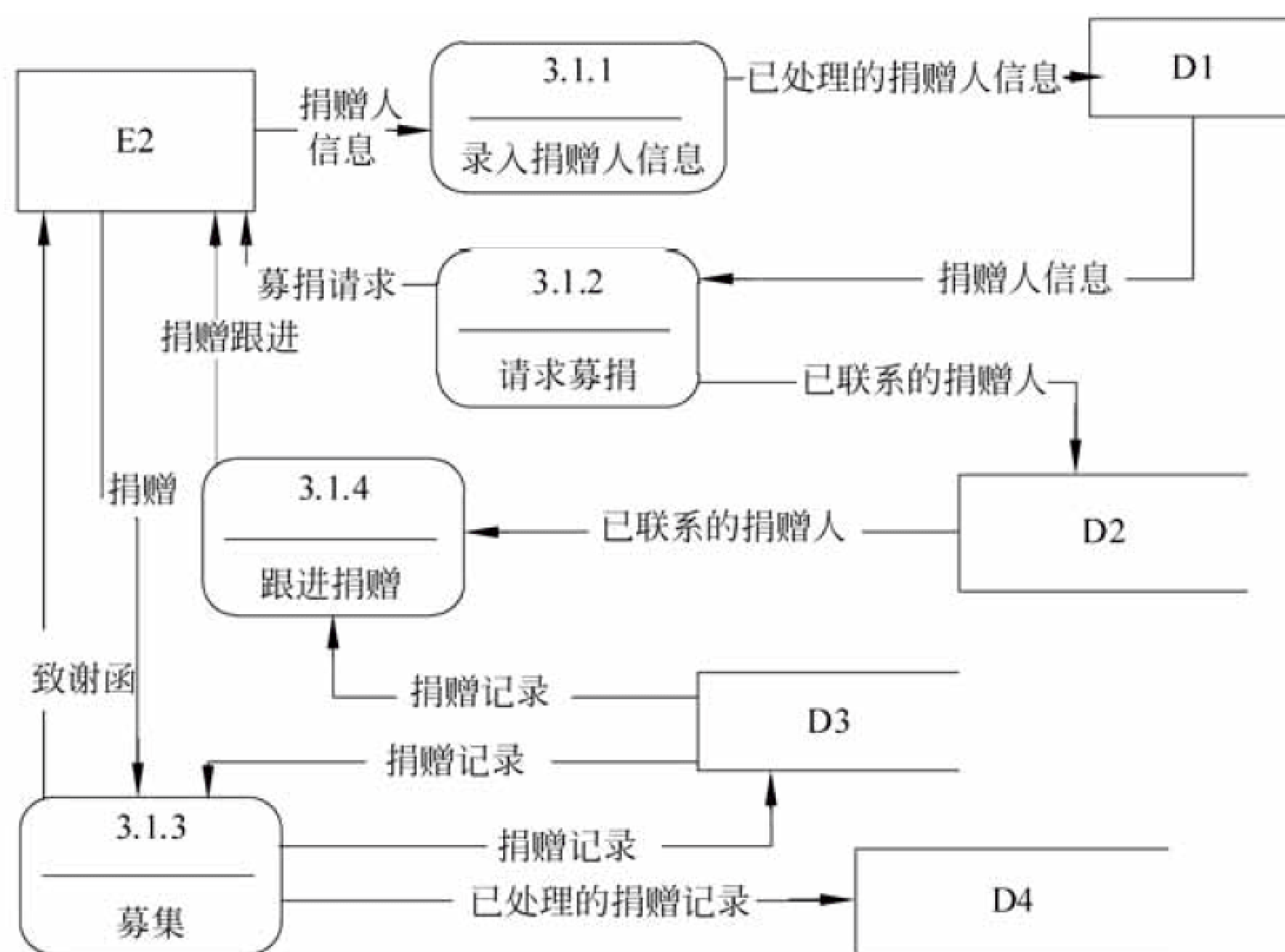


图 1-3 2 层数据流图

在建分层 DFD 时，根据需求情况可以将数据存储建模在不同层次的 DFD 中，注意在绘制下层数据流图时要保持父图与子图平衡。父图中某加工的输入输出数据流必须与它的子图的输入输出数据流在数量和名称上相同，或者父图中的一个输入（或输出）数据流对应于子图中几个输入（或输出）数据流，而子图中组成这些数据流的数据项全体正好是父图中的这一个数据流。

【问题 1】

本问题给出 0 层 DFD，要求根据描述确定图中的外部实体。分析题目中描述，并结合已在图中给出的数据流进行分析。从题目的说明中可以看出，与系统交互实体包括志愿者、捐赠人、募款机构和场馆，这四个作为外部实体。

对应图 1-1 中数据流和实体的对应关系，可知 E1 为志愿者，E2 为捐赠人，E3 为募款机构，E4 为场馆。

【问题 2】

本题考查分层 DFD 的加工分解，以及父图与子图的平衡。图 1-2 中对图 1-1 的加工 3 进行进一步分解，根据说明（3）中对加工 3 的描述对图 1-2 进行分析。首先需要确定活动时间范围，其输入数据流是活动请求，输出流为活动时间。然后是搜索场馆，其输入流为活动时间，输出活动时间和地点，同时向场馆发送的场馆可用性请求和获得的场

馆可用性分别作为输入和输出数据流。在确定活动时间和地点的基础上推广募捐活动，活动时间和地点是其输入流，活动信息作为其输出流，流向举办活动并募集资金，从募款机构获取资金并向其发放赠品，加工 2 收集募得的资金和物品，因此 3.5 还需要将所募集资金作为输出流。获取和处理捐赠（资金和物品）时以捐赠请求作为其输入流，输出流为所募集的捐赠，因为既有资金又有物品，而从募款机构募得的只有资金，将图 1-1 中加工 3 流向加工 2 的数据流，分为所募集资金和所募集物品，而 3.5 的输出流中只有所募集资金。

因此，P1 为确定活动时间范围，P2 为搜索场馆，P3 为推广募捐活动。图 1-2 中缺失了从 2 到 3.3 的活动时间和从 3.5 到 2 的所募集资金这两条数据流。

题目给出处理和捐赠人之间的交互进一步描述，对 3.1 进一步建模下层数据流图（图 1-3）。分解加工 3.1，确定相关数据流。其中根据加工 2 的捐赠请求进行募集，所募捐赠需要返回给加工 2。

根据父图与子图的平衡原则，图 1-3 中此处也缺失了捐赠请求和所募集资金和所募集物品。

【问题 3】

本问题考查 2 层 DFD 中数据存储的确定。本案例中，数据存储的描述都是在这一部分描述给出，所以数据存储建模在此层体现。对应说明可知，D1 为捐赠人信息表，D2 为以联系的捐赠人表，D3 为捐赠表，D4 为已处理捐赠表。

参考答案

【问题 1】

E1：志愿者　　E2：捐赠人　　E3：募款机构　　E4：场馆

【问题 2】

P1：确定活动时间范围

P2：搜索场馆

P3：推广募捐活动

数据流名称	起点	终点
所募集资金	3.5 或 举办活动并募集资金	2
活动请求	2	3.2 或 确定活动时间范围
捐赠请求	2	3.1.3 募集
所募集捐赠	3.1.3 或 募集	2
或		
所募集资金	3.1.3 或 募集	2
所募集物品	3.1.3 或 募集	2

注：数据流没有次序要求；表中 2 处可以是“确定募捐需求收集所募捐赠”。

【问题 3】

D1：捐赠人信息表　　D2：已联系的捐赠人表

D3：捐赠表　　D4：已处理捐赠表

试题二（共 15 分）

阅读下列说明，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某电视台拟开发一套信息管理系统，以方便对全台的员工、栏目、广告和演播厅等进行管理。

【需求分析】

（1）系统需要维护全台员工的详细信息、栏目信息、广告信息和演播厅信息等。员工的信息主要包括：工号、姓名、性别、出生日期、电话、住址等。栏目信息主要包括：栏目名称、播出时间、时长等。广告信息主要包括：广告编号、价格等。演播厅信息包括：房间号、房间面积等。

（2）电视台根据调度单来协调各档栏目、演播厅和场务。一销售档栏目只会占用一个演播厅，但会使用多名场务来进行演出协调。演播厅和场务可以被多个栏目循环使用。

（3）电视台根据栏目来插播广告。每档栏目可以插播多条广告，每条广告也可以在多档栏目插播。

（4）一档栏目可以有多个主持人，但一名主持人只能主持一档栏目。

（5）一名编辑人员可以编辑多条广告，一条广告只能由一名编辑人员编辑。

【概念模型设计】

根据需求阶段收集的信息设计的实体联系图（不完整）如图 2-1 所示。

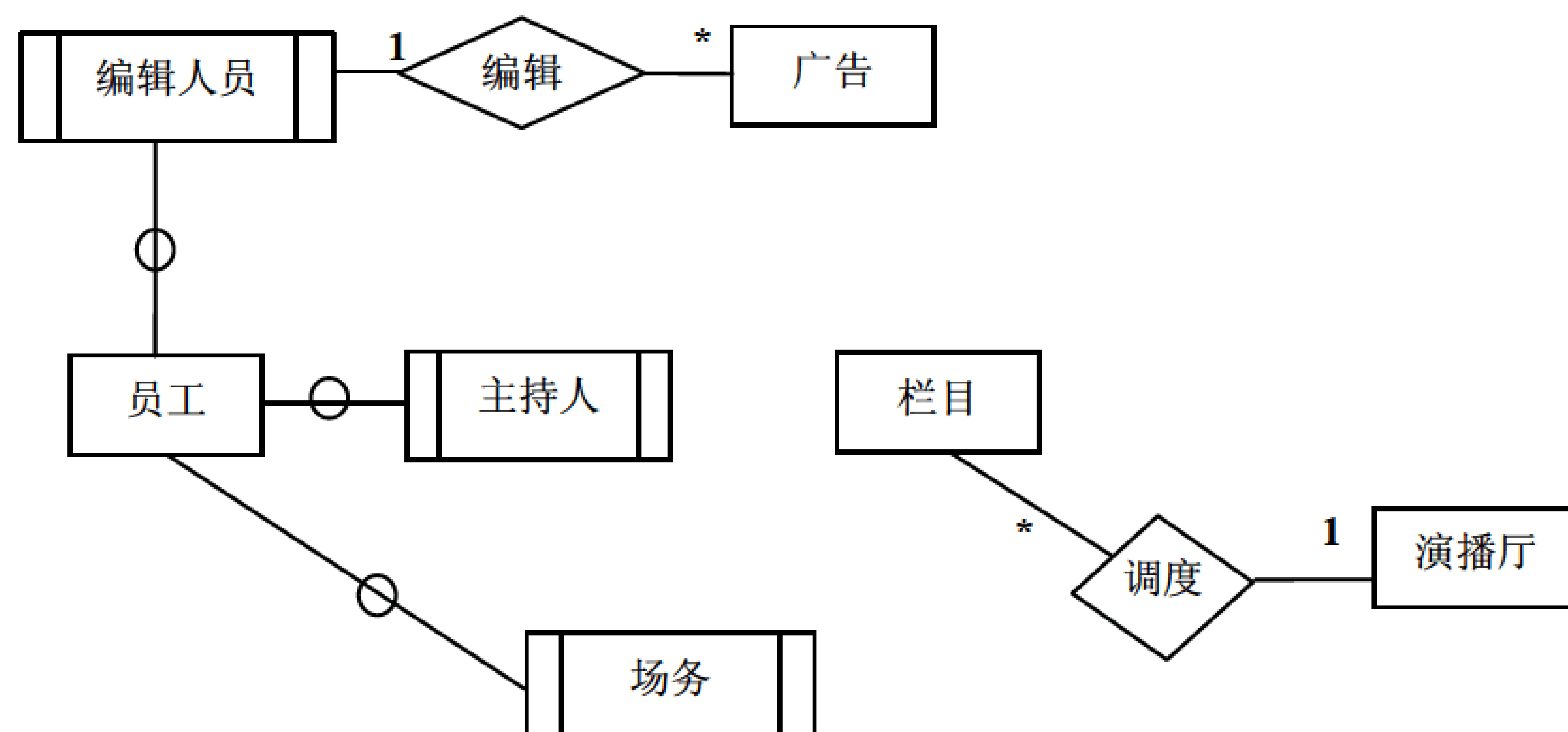


图 2-1　实体联系图

【逻辑结构设计】

根据概念模型设计阶段完成的实体联系图，得出如下关系模式（不完整）：

演播厅（<u>房间号</u>，房间面积）

栏目（<u>栏目名称</u>，播出时间，时长）

广告（广告编号，销售价格，____（1）____）

员工（<u>工号</u>，姓名，性别，出生日期，电话，住址）

主持人（主持人工号，____（2）____）

插播单（____（3）____，播出时间）

调度单（____（4）____）

【问题 1】（7 分）

补充图 2-1 中的联系和联系的类型。

【问题 2】（5 分）

根据图 2-1，将逻辑结构设计阶段生成的关系模式中的空（1）～（4）补充完整，并用下划线指出（1）～（4）所在关系模式的主键。

【问题 3】（3 分）

现需要记录广告商信息，增加广告商实体。一个广告商可以提供多条广告，一条广告只由一个广告商提供。请根据该要求，对图 2-1 进行修改，画出修改后的实体间联系和联系的类型。

试题二分析

本题考查数据库设计，属于比较传统的题目，考查点也与往年类似。

【问题 1】

本问题考查数据库的概念结构设计，题目要求补充完整实体联系图中的联系和联系的类型。

根据题目的需求描述可知，一个栏目可以插播多条广告，而多条广告也可以在多个栏目中播放，因此栏目和广告之间存在“插播”联系，联系的类型为多对多（*:*，或 m:n）。

根据题目的需求描述可知，一个栏目可以有多个主持人，而一个主持人只能主持一档栏目，因此栏目和主持人之间存在“主持”联系，联系的类型为一对多（1:*，或 1:n）。

根据题目的需求描述可知，一个栏目需要使用多名场务来进行演出协调，场务可以被多个栏目循环使用，因此演播厅、栏目和场务之间存在“调度”联系，联系的类型为 1 对多对多（1:*:*，或 1:m:n）。

【问题 2】

本问题考查数据库的逻辑结构设计，题目要求补充完整各关系模式，并给出各关系模式的主键。

根据实体联系图和需求描述，广告记录广告编号、销售价格和编辑人员工号。所以，对于“广告”关系模式，需补充属性“广告编号”。广告编号为广告的主键。

根据实体联系图和需求描述，主持人记录主持人工号和所属的栏目名称。所以，对

于“主持人”关系模式，需补充属性“主持人工号”。主持人工号为主持人的主键。

根据实体联系图和需求描述，插播单需要记录栏目名称、广告编号和播出的时间。所以，对于“插播单”关系模式，需补充属性“栏目名称”和“广告编号”。栏目名称和广告编号联合作为插播单的主键。

根据实体联系图和需求描述，调度单需要记录栏目名称、房间号和参与的场务工号。所以，对于“调度单”关系模式，需补充属性“栏目名称”“房间号”和“场务工号”。栏目名称、房间号和场务工号联合作为插播单的主键。

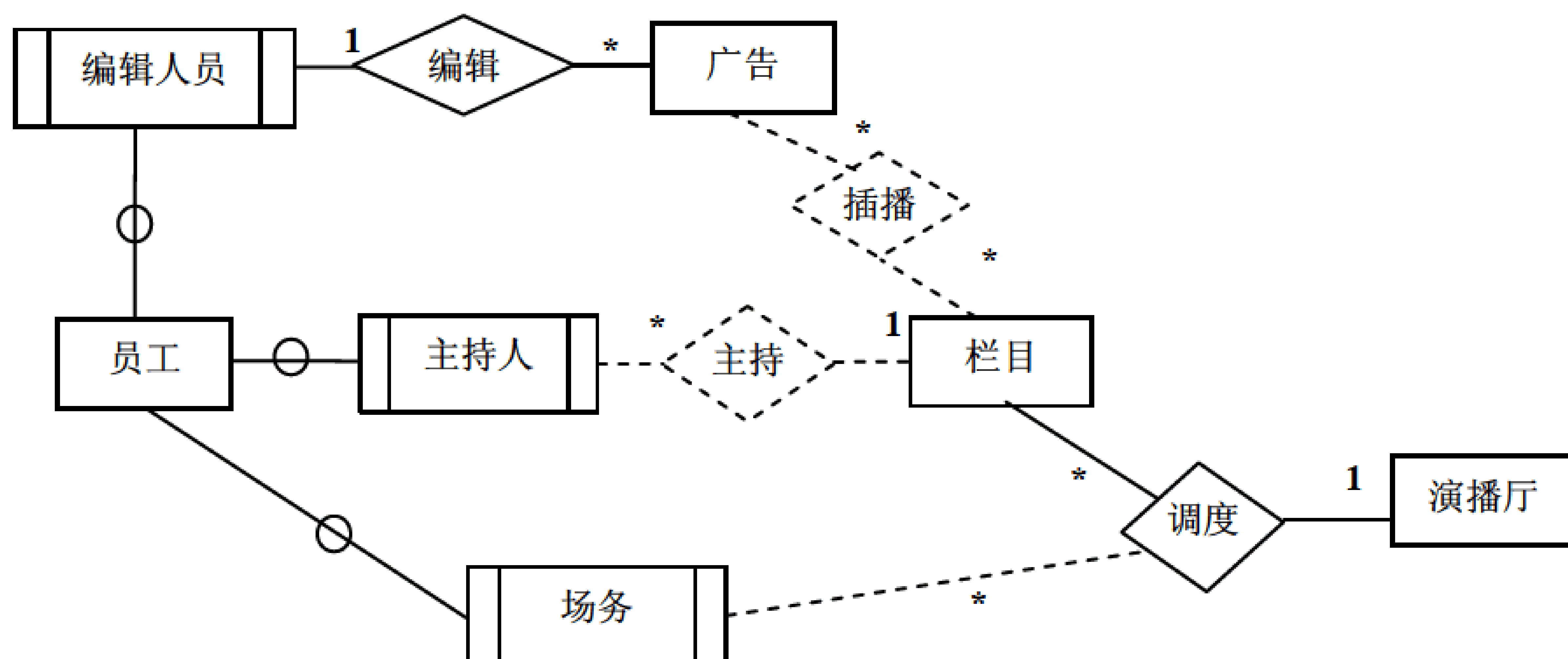
【问题3】

本问题考查数据库的概念结构设计，根据新增的需求增加实体联系图中的实体的联系和联系的类型。

根据问题描述，一个广告商可以提供多条广告，一条广告只由一个广告商提供。则须在广告商实体和广告实体之间存在“提供”联系，联系的类型为1对多（1:*，或1:n）。

参考答案

【问题1】



说明：*填写为m和n均可。

【问题2】

广告（<u>广告编号</u>，销售价格，编辑人员工号）

主持人（<u>主持人工号</u>，栏目名称）

插播单（<u>栏目名称，广告编号</u>，播出时间）

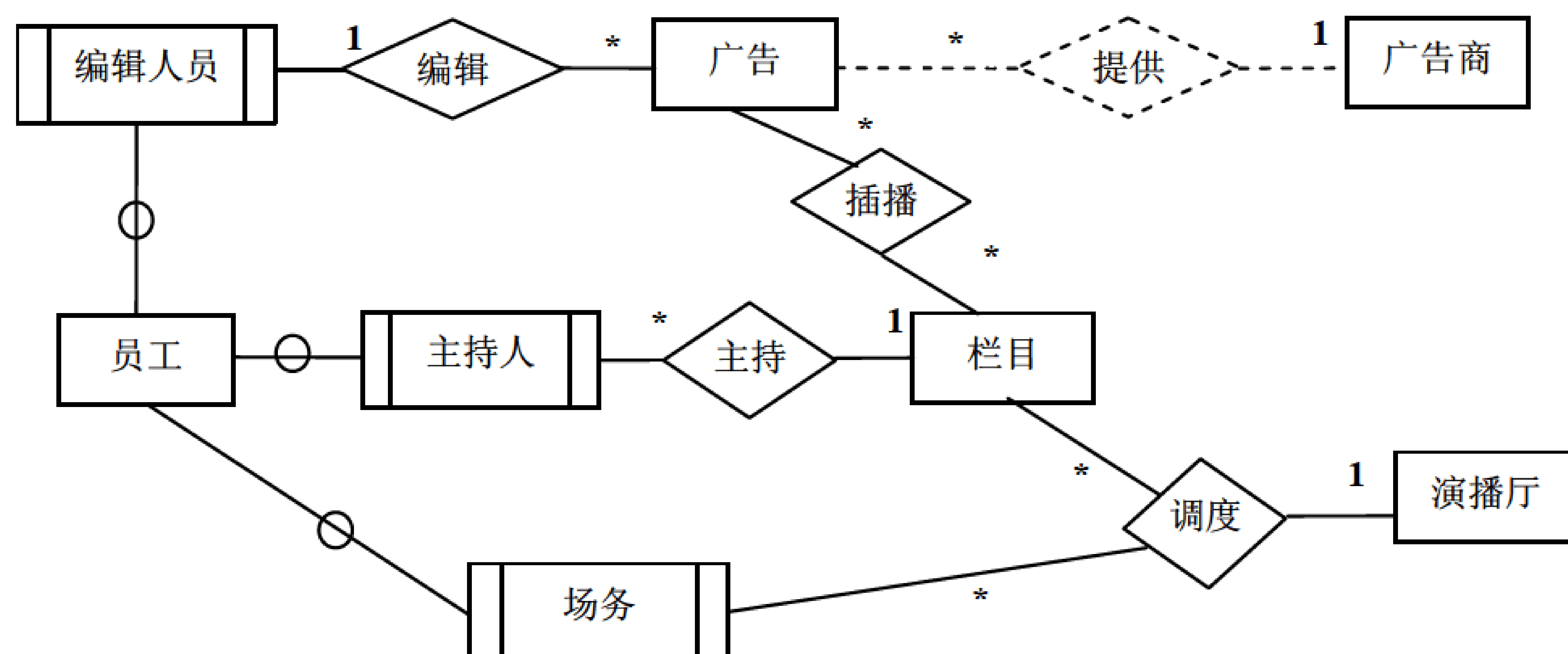
调度单（<u>栏目名称，房间号，场务工号</u>）

【问题3】

说明：*填写为m和n均可。

试题三（共15分）

阅读下列说明和图，回答问题1至问题3，将解答填入答题纸的对应栏内。



【说明】

某城市拟开发一个基于 Web 的城市黄页，公开发布该城市重要的组织或机构（以下统称为客户）的基本信息，方便城市生活。该系统的主要功能描述如下：

（1）搜索信息：任何使用 Internet 的网络用户都可以搜索发布在城市黄页中的信息，例如客户的名称、地址、联系电话等。

（2）认证：客户若想在城市黄页上发布信息，需通过系统的认证。认证成功后，该客户成为系统授权用户。

（3）更新信息：授权用户登录系统之后，可以更改自己在城市黄页中的相关信息，例如变更联系电话等。

（4）删除客户：对于拒绝继续在城市黄页上发布信息的客户，由系统管理员删除该客户的相关信息。

系统采用面向对象方法进行开发，在开发过程中认定出如表 3-1 所示的类。系统的用例图和类图分别如图 3-1 和 3-2 所示。

表 3-1　类列表

类名	说明
InternetClient	网络用户
CustomerList	客户集，维护城市黄页上的所有客户信息
Customer	客户信息，记录单个客户的信息
RegisteredClient	授权用户
Administrator	系统管理员

【问题 1】（5 分）

根据说明中的描述，给出图 3-1 中 A1 和 A2 处所对应的参与者、UC1 和 UC2 处所对应的用例以及（1）处的关系。

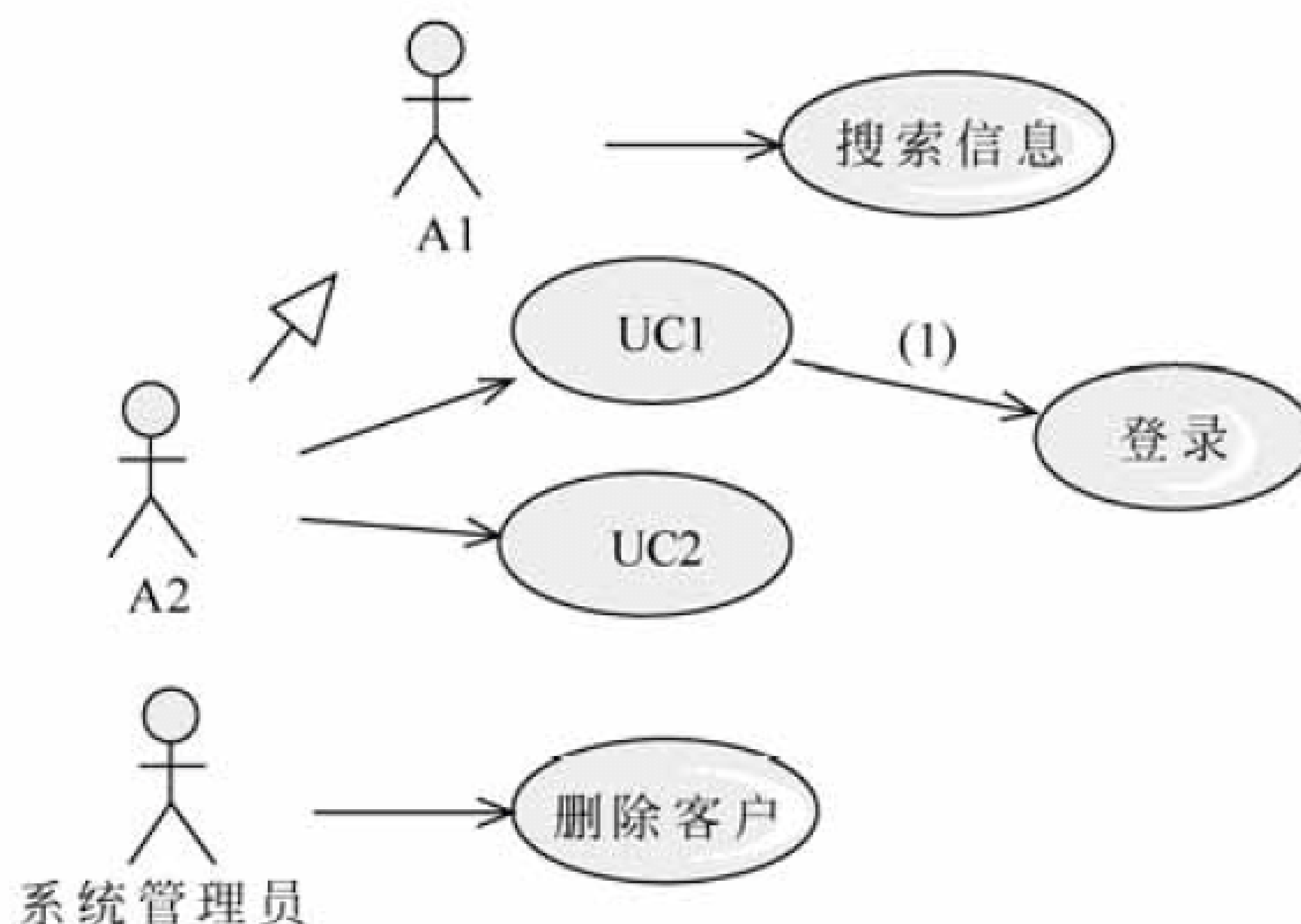


图 3-1　系统用例图

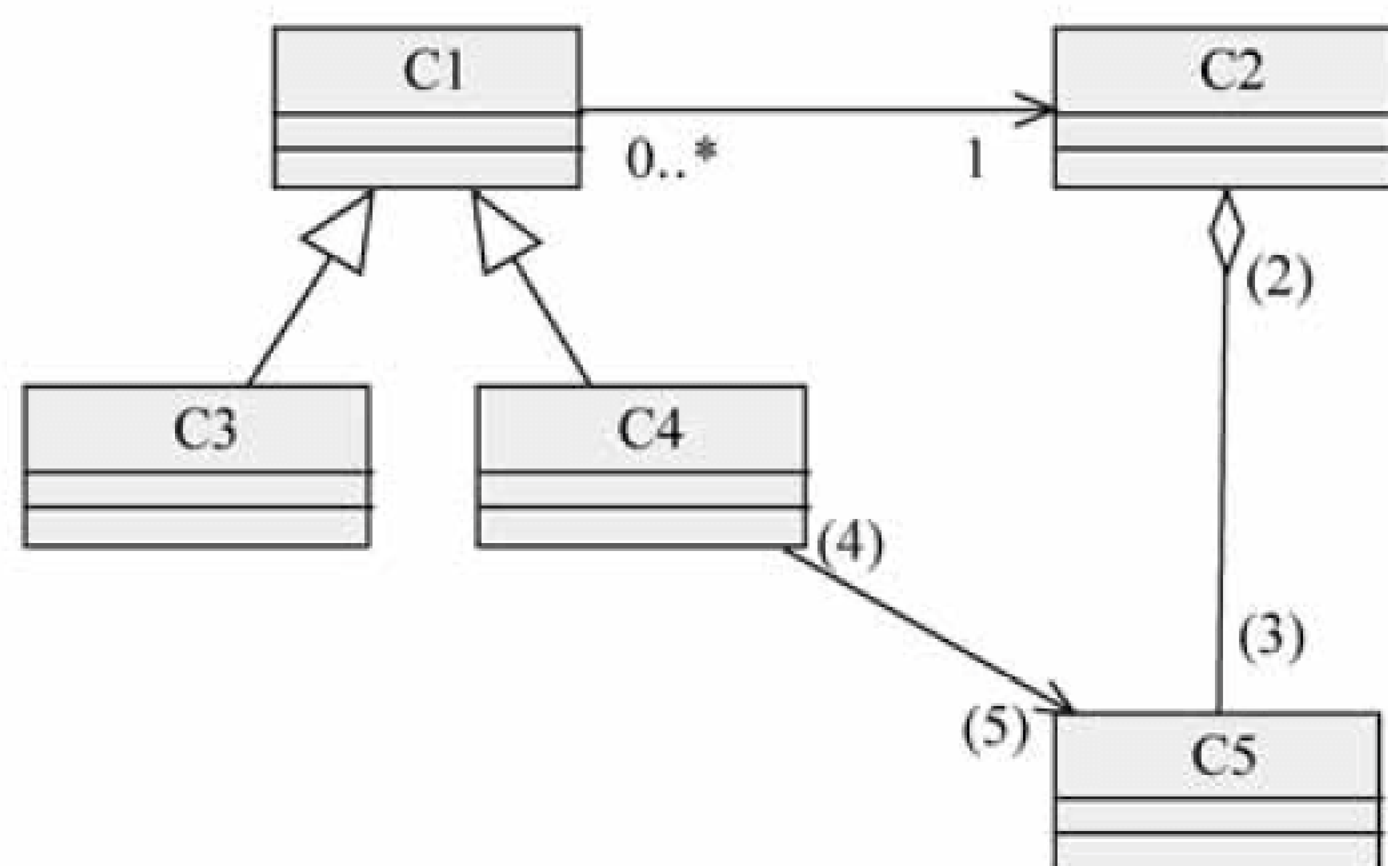


图 3-2　系统类图

【问题 2】（7 分）

根据说明中的描述，给出图 3-2 中 C1～C5 所对应的类名（表 3-1 中给出的类名）和（2）～（5）处所对应的多重度。

【问题 3】（3 分）

认定类是面向对象分析中非常关键的一个步骤。一般首先从问题域中得到候选类集合，再根据相应的原则从该集合中删除不作为类的，剩余的就是从问题域中认定出来的类。简要说明选择候选类的原则，以及对候选类集合进行删除的原则。

试题三分析

本题属于经典的考题，主要考查面向对象分析方法以及 UML 的用例图和类图的相关应用。

【问题 1】

本问题要求将图 3-1 所给出的用例图补充完整。用例图的构成要素有：参与者

（Actor）、用例（Usecase）以及用例之间的关系。

本题背景描述简单，再结合图 3-1 中给出的两个用例“搜索信息”和“删除客户”，很容易确定出“认证”“更新信息”就是需要补充的用例。下面只需要确定参与者 A1、A2 以及这两个用例与 A1、A2 之间的关系即可将图补充完整。在说明中出现了两类角色：客户和系统管理员。系统管理员已经作为参与者明确标识在 3-1 中了。那么 A1 和 A2 只能从“客户”这个角色中产生。明显地，“客户”在说明中被细分成了两类：网络用户和授权用户，而认证用户是一种特殊的网络用户。因此，A1 应该对应“网络用户”，A2 应该对应“授权客户”。要成为授权用户，必须首先经过认证。成为授权用户之后，不仅可以搜索信息，还具有更新信息的权限。更新信息时要求首先登录系统，因此“登录”是“更新信息”操作中所包含的一个必须步骤。由此可以确定 UC1 对应“更新信息”，UC2 对应“认证”，而（1）处的关系应该是<<include>>。

【问题 2】

本问题考查类建模。表 3-1 已经给出了类列表，这样对照图 3-2 寻找对应关系即可。图 3-2 中有两个明显的结构：继承（C1、C3 和 C4 之间）和聚集（C2 和 C5 之间）。

先确定聚集关系相关的类。由表 3-1 可以找出两个明显具有部分-整体概念的类：CustomerList 和 Customer，由类的说明可以看出，CustomerList 表示整体概念，Customer 表示部分概念。由此可以确定 C2 对应 CustomerList，C5 对应 Customer。同时可以确定出这两个类之间的多重度，即（2）处应为 1，（3）处为 0..*。

下面来确定继承关系相关的类。图 3-1 中已经出现了一个继承关系（A1 和 A2 之间），这就给出了一个明显的提示：图 3-2 中的继承关系与这两类角色相关。回到表 3-1 中，发现了 3 个与角色相关的类：InternetClient（网络用户）、RegisteredClient（授权用户）和 Administrator（系统管理员）。由于已经确定了 C5 是 Customer，而能够对 Customer 进行操作的只有系统管理员。因此具有继承关系的这 3 个类应分别是：C1-网络用户，C3-授权用户，C4-系统管理员。（4）、（5）处的多重度也可以确定下来，均为 0..1。

【问题 3】

本问题考查面向对象分析过程中认定类/对象的过程。通常分为两个步骤进行，首先将问题域（需求分析）中所有自然存在的名词都选出来，构成候选类集合。然后针对这个候选类集合，将满足以下原则的名词从候选类集合中删除：含义相近（冗余）、含义不明确的对象、暗示实现方式的、表示属性或特征、有动词含义的名词（表示行为和方法）。最后剩余的就是所认定的类/对象。

参考答案

【问题 1】

A1：网络用户　或　InternetClient　　A2：授权用户　或　RegisteredClient

UC1：更新信息　　UC2：认证

（1）：<<include>>

【问题 2】

C1：InternetClient　　C2：CustomerList　　C3：Administrator

C4：RegisteredClient　C5：Customer

（2）1　　（3）0..*　　（4）0..1　　（5）0..1

【问题 3】

选择候选类时通常考虑的是问题域中自然存在的名词。

具有下列特征的候选类需要删除：含义相近（冗余）、含义不明确的对象、暗示实现方式的、表示属性或特征、有动词含义的名词（表示行为和方法）。

试题四（共 15 分）

阅读下列说明和 C 代码，回答问题 1 至问题 3，将解答写在答题纸的对应栏内。

【说明】

设有 m 台完全相同的机器运行 n 个独立的任务，运行任务 i 所需要的时间为 t_i，要求确定一个调度方案，使得完成所有任务所需要的时间最短。

假设任务已经按照其运行时间从大到小排序。算法基于最长运行时间作业优先的策略：按顺序先把每个任务分配到一台机器上，然后将剩余的任务依次放入最先空闲的机器。

【C 代码】

下面是算法的 C 语言实现。

（1）常量和变量说明

m：机器数

n：任务数

t[]：输入数组，长度为 n，其中每个元素表示任务的运行时间，下标从 0 开始

s[][]：二维数组，长度为 m*n，下标从 0 开始，其中元素 s[i][j]表示机器 i 运行的任务 j 的编号

d[]：数组，长度为 m，其中元素 d[i]表示机器 i 的运行时间，下标从 0 开始

count[]：数组，长度为 m，下标从 0 开始，其中元素 count[i]表示机器 i 运行的任务数

i：循环变量

j：循环变量

k：临时变量

max：完成所有任务的时间

min：临时变量

（2）函数 schedule

```
void schedule(){
    int i, j, k, max = 0;
```

```
        for(i = 0; i < m; i++){
            d[i] = 0;
            for(j = 0; j < n; j++){
                s[i][j] = 0;
            }
        }

        for(i = 0; i < m; i++){                //分配前 m 个任务
            s[i][0] = i;
            ____(1)____;
            count[i] = 1;
    }

        for(____(2)____; i < n; i++ ){         //分配后 n-m 个任务
            int min = d[0];
            k = 0;
            for ( j = 1; j < m; j++ ){         //确定空闲机器
                if ( min > d[j] ){
                    min = d[j];
                    k = j;                     //机器 k 空闲
                }
            }
            ____(3)____;
            count[k] = count[k] + 1;
            d[k] = d[k] + t[i];
    }

        for( i = 0; i < m; i++ ){              //确定完成所有任务需要的时间
            if ( ____(4)____ ){
                max = d[i];
            }
        }
    }
```

【问题 1】（8 分）

根据说明和 C 代码，填充 C 代码中的空（1）～（4）。

【问题 2】（2 分）

根据说明和 C 代码，该问题采用了____（5）____算法设计策略，时间复杂度为____（6）____（用 O 符号表示）。

【问题 3】（5 分）

考虑实例 m = 3（编号 0～2），n = 7（编号 0～6），各任务的运行时间为{16,14,6,5,4,3,2}。则在机器 0、1 和 2 上运行的任务分别为 （7） 、 （8） 和 （9） （给出任务编号）。从任务开始运行到完成所需要的时间为 （10） 。

试题四分析

本题考查算法设计与分析技术以及算法的 C 语言实现，是比较传统的题目，要求考生细心分析题目中所描述的内容。

【问题 1】

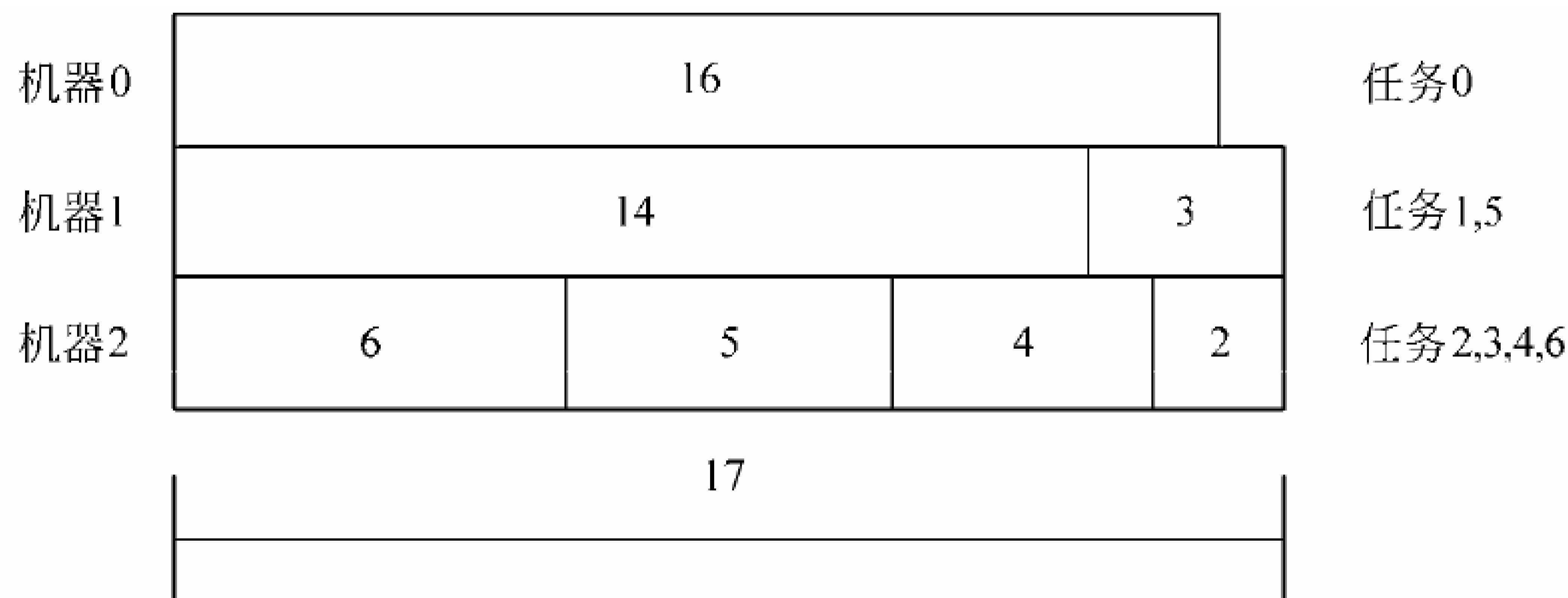
根据题中说明和代码注释，算法首先初始化数组 d 和 s 中的元素；然后将 m 个任务分配到 m 台机器上，此时将任务 0,1,⋯,m-1 分别分配到机器 0,1,⋯,m-1 上，同时设置 d、s 和 count 数组中的相关元素的值，故空格（1）填写 d[i] = t[i]；接下来将剩下的 n-m 个任务分配到 m 台机器上，从任务 m 开始，因此空格（2）填写 i = m，确定首先空闲的机器 k，将当前尚未分配的第一个任务分配到机器 k 上，并设置 d、s 和 count 数组中的相关元素的值，故空格（3）填写 s[k][count[k]] = i；最后确定从任务开始到结束所需要的时间，从所有机器的运行时间中选择运行时间最长的机器的运行时间，即最大的 d[i]，因此空格（4）填写 max < d[i]。

【问题 2】

根据上述 C 代码，算法有两处两重循环，时间复杂度为 O(mn)，有两处一重循环，时间复杂度为 O(m)，因此整个算法的时间复杂度为 O(mn)。

【问题 3】

根据说明中的算法思想和 C 代码，首先将任务 0、1 和 2 分配到机器 0、1 和 2 上运行，运行时间分别为 16、14 和 6。由于任务 2 的时间最短，故任务 3 在机器 2 上运行，机器 2 上的运行时间为 6+5=11，仍然是时间最短，任务 4 继续在机器 2 上运行，机器 2 上的运行时间为 11+4=15。此时机器 1 上的运行时间最短，任务 5 在机器 1 上运行，机器 1 上的运行时间为 14+3=17。此时机器 2 上的运行时间最短，任务 6 在机器 2 上运行，机器 2 上的运行时间为 15+2=17。所以任务分配完成。根据此分配，在机器 0 上运行任务 0，运行时间为 16；机器 1 上运行任务 1 和 5，运行时间为 17；在机器 2 上运行任务 2，3，4 和 6，运行时间为 17。因此从任务开始到结束的时间为 17。



参考答案

【问题 1】

（1） d[i] = t[i]　（2） i = m　（3） s[k][count[k]] = i　（4） max < d[i]

【问题 2】

（5） 贪心　（6） O(mn)

【问题 3】

（7） 0　（8） 1,5　（9） 2,3,4,6　（10） 17

试题五（共 15 分）

阅读下列说明和 C++代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

现要求实现一个能够自动生成求职简历的程序。简历的基本内容包括求职者的姓名、性别、年龄及工作经历等。希望每份简历中的工作经历有所不同，并尽量减少程序中的重复代码。

现采用原型（Prototype）模式来实现上述要求，得到如图 5-1 所示的类图。

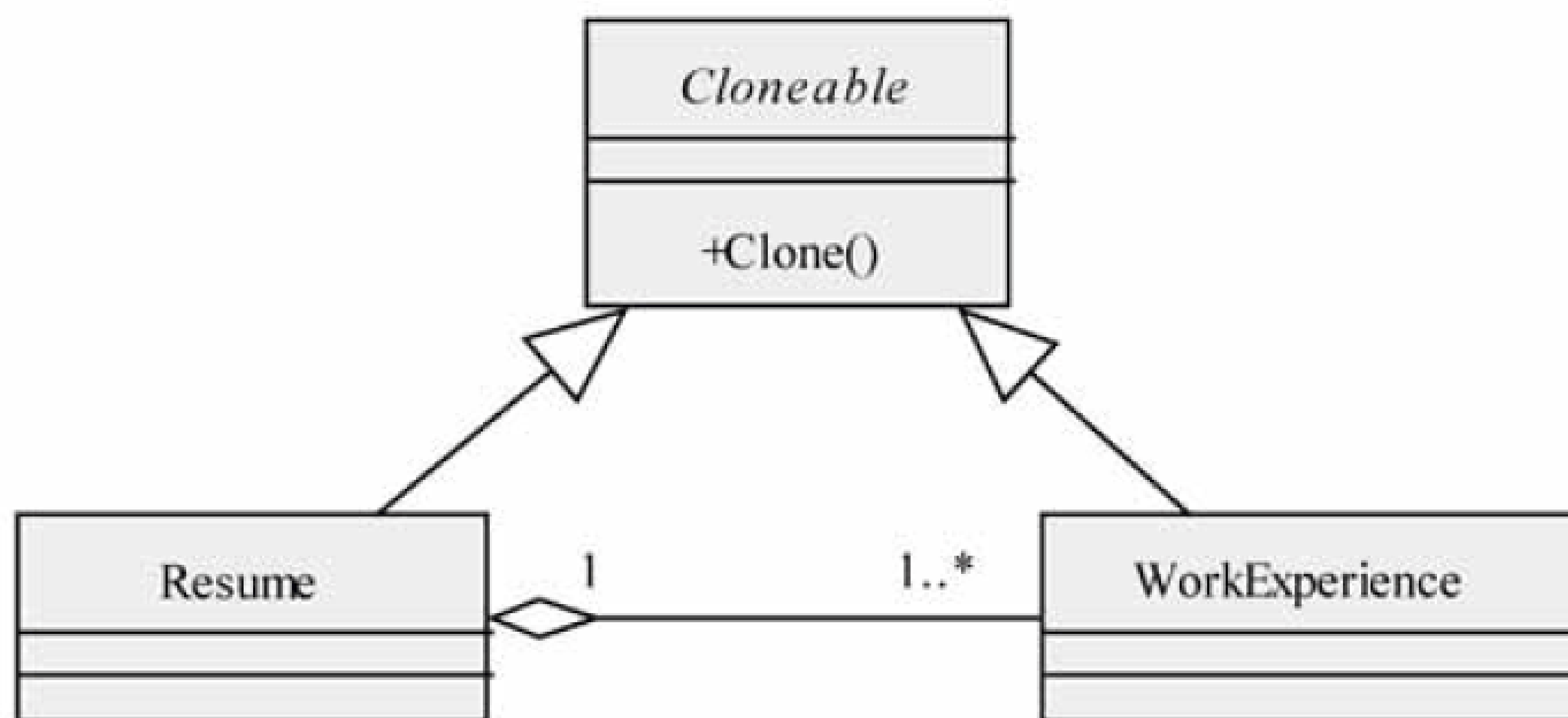


图 5-1　类图

【C++代码】

```
#include <string>
using namespace std;
class Cloneable {
public:
___________(1)___________;
};
class WorkExperience : public Cloneable {   // 工作经历
private:
   string workDate;
```

```
    string company;
public:
    Cloneable* Clone() {
        ______(2)______;
        obj->workDate = this->workDate;
        obj->company = this->company;
        return obj;
    }
    // 其余代码省略
};
class Resume : public Cloneable {     // 简历
private:
    string name; string sex; string age;
    WorkExperience* work;
    Resume(WorkExperience* work) {
this->work = ______(3)______;
    }
public:
    Resume(string name) { /* 实现略 */   }
    void SetPersonalInfo(string sex, string age) { /* 实现略 */   }
    void SetWorkExperience(string workDate, string company) { /* 实现略 */ }
    Cloneable* Clone () {
        ______(4)______;
        obj->name = this->name;
        obj->sex = this->sex;
        obj->age = this->age;
        return obj;
    }
};
int main() {
Resume *a = new Resume("张三");
a->SetPersonalInfo("男", "29");
a->SetWorkExperience("1998~2000", "XXX 公司");
Resume *b = ______(5)______;
a->SetWorkExperience("2001~2006", "YYY 公司");
return 0;
}
```

试题五分析

本题考查原型（Prototype）模式的概念及应用。

Prototype 模式是一种对象创建型模式。Prototype 模式通过给出一个原型对象来指明

所要创建的对象的类型，然后通过复制这个原型对象的方法，创建出更多同类型的对象。原型模式又可以分为两种：浅克隆和深克隆。浅克隆仅仅复制所考虑的对象，而不复制它所引用的对象，也就是其中的成员对象并不复制；深克隆除了对象本身被复制外，对象包含的引用也被复制，即成员对象也被复制。

原型模式的优点是：

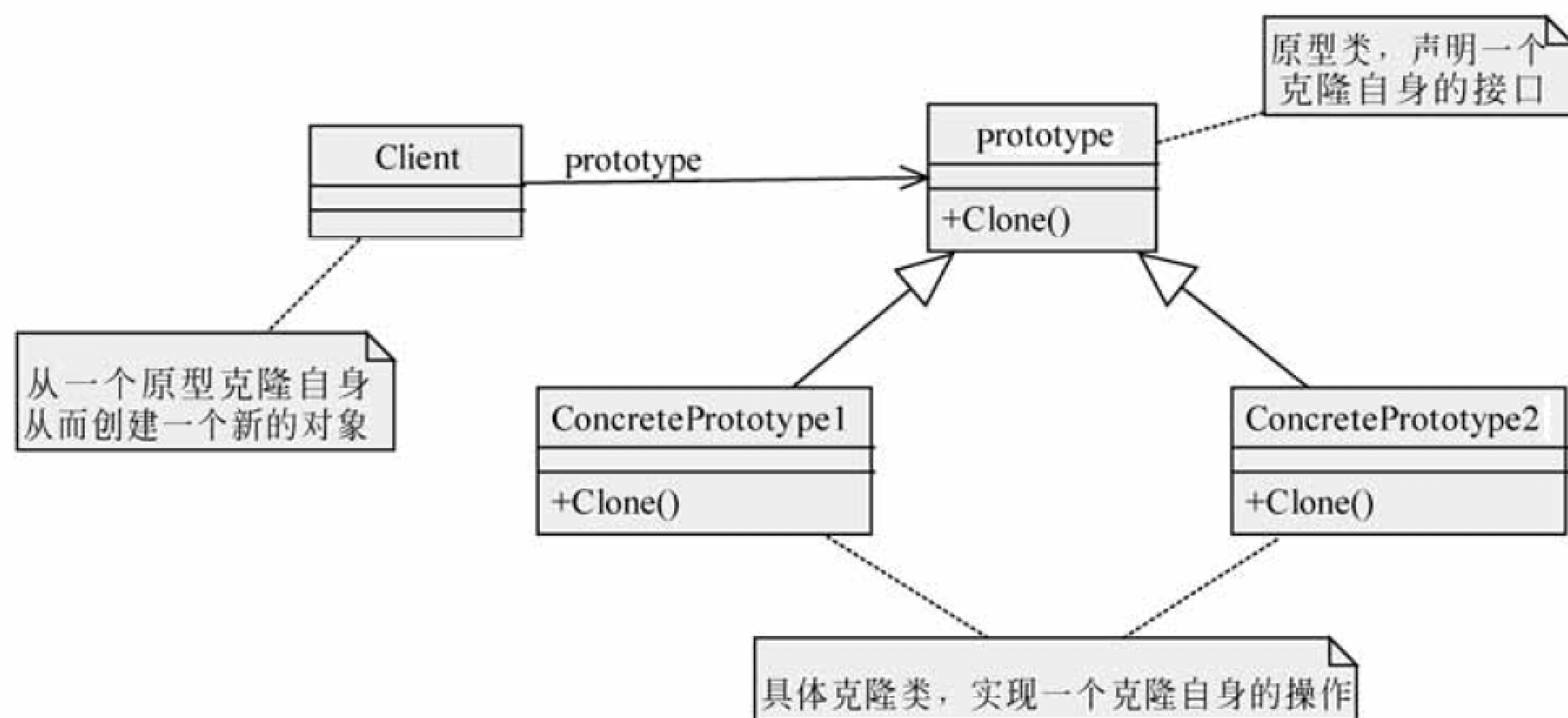
（1）向客户隐藏制造新实例的复杂性；

（2）提供让客户能够产生未知类型对象的选项；

（3）在某些环境下，复制对象比创建新对象更有效。

在一个复杂的类层次中，当系统必须从其中的许多类型创建新对象时，可以考虑原型模式。使用原型模式的缺点是：对象的复制有时相当复杂。

Prototype 模式的结构如下图所示。



题目利用原型模式来实现自动生成多份求职简历，而且每份简历的工作经历有所差别。题目中使用的是原型模型中的深克隆。题目中的类 Cloneable 对应上图中的类 Prototype，Resume 和 WorkExperience 分别是两个具体克隆类。由于 WorkExperience 是 Resume 的一部分，所以在 Resume 中有一个 WorkExperience 类型的成员对象。

下面来分析程序。

第（1）空出现在类 Cloneable 的定义中。类 Cloneable 的作用是声明一个克隆自身的接口，在 C++中通常都采用抽象类来定义这种抽象操作接口。C++中的抽象类是包含了至少一个纯虚拟函数的类。纯虚拟函数的语法是：

virtual <函数名>(<参数列表>) =0;

下面来确定纯虚拟函数的原型。由类图 5-1 以及 Cloneable 的子类 WorkExperience 中定义的成员函数 Clone()可知，（1）空应该填写 virtual Cloneable* Clone() = 0。

空（2）出现在类 WorkExperience 成员函数 Clone()的定义中。WorkExperience 作为具体克隆类，应实现继承自父类的纯虚拟函数 Clone()。根据已给出的代码的提示，在 Clone()方法中，应再创建一个 WorkExperience 对象，而这个对象的名称就是 obj。因此（2）空处应填 WorkExperience *obj = new WorkExperience()。

本题采用的是深克隆，除了对象本身被复制外，成员对象也被复制。work 是 Resume 中的成员对象，所以（3）空是对成员对象的克隆，因此应调用 work 对象的克隆操作，即(WorkExperience *)work->Clone()。

第（4）空出现在 Resume 的 Clone()方法中，功能与第（2）空类似，完成 Resume 对象的克隆，所以第（4）空应填 Resume *obj = new Resume(this->work)。

第（5）空是对原型模式的应用。用简历 a 克隆出简历 b，即调用 a 的克隆操作，所以第（5）空应填(Resume *)a->Clone()。

参考答案

（1）virtual Cloneable* Clone() = 0

（2）WorkExperience *obj = new WorkExperience()

（3）(WorkExperience *)work->Clone()

（4）Resume *obj = new Resume(this->work)

（5）(Resume *)a->Clone()

试题六（共 15 分）

阅读下列说明和 Java 代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

现要求实现一个能够自动生成求职简历的程序。简历的基本内容包括求职者的姓名、性别、年龄及工作经历等。希望每份简历中的工作经历有所不同，并尽量减少程序中的重复代码。

现采用原型（Prototype）模式来实现上述要求，得到如图 6-1 所示的类图。

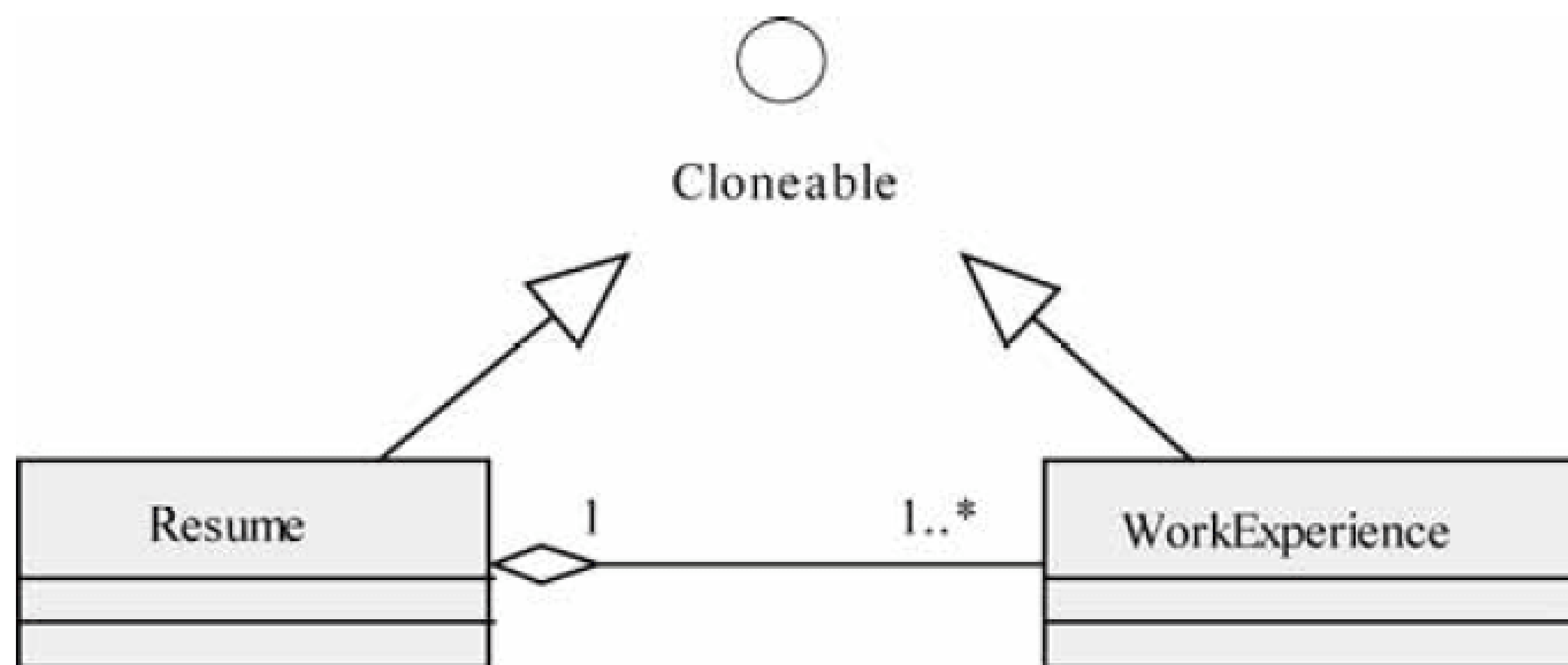


图 6-1 类图

【Java 代码】

```
class WorkExperience ____(1)____ Cloneable {   // 工作经历
  private String workDate;
  private String company;
  public Object Clone() {
     ____(2)____;
     obj.workDate = this.workDate;
     obj.company = this.company;
     return obj;
  }
}

class Resume ____(3)____ Cloneable {   // 简历
  private String name;
  private String sex;
  private String age;
  private WorkExperience work;

  public Resume(String name) {
    this.name = name;    work = new WorkExperience();
  }

  private Resume(WorkExperience work) {
    this.work =____(4)____;
  }

  public void SetPersonalInfo(String sex, String age) {   /* 代码略 */ }
  public void SetWorkExperience(String workDate, String company) {   /* 代
码略 */  }
  public Object Clone () {
     Resume obj = ____(5)____;
     // 其余代码省略
     return obj;
  }
}

class WorkResume {
  public static void main(String[] arg) {
    Resume a = new Resume("张三");
    a.SetPersonalInfo("男", "29");
    a.SetWorkExperience("1998~2000", "XXX 公司");
    Resume b = ____(6)____;
```

```
        b.SetWorkExperience("2001~2006", "YYY 公司");
    }
}
```

试题六分析

本题考查原型（Prototype）模式的概念及应用。

Prototype 模式是一种对象创建型模式。Prototype 模式通过给出一个原型对象来指明所要创建的对象的类型，然后通过复制这个原型对象的方法，创建出更多同类型的对象。原型模式又可以分为两种：浅克隆和深克隆。浅克隆仅仅复制所考虑的对象，而不复制它所引用的对象，也就是其中的成员对象并不复制；深克隆除了对象本身被复制外，对象包含的引用也被复制，即成员对象也被复制。

原型模式的优点是：

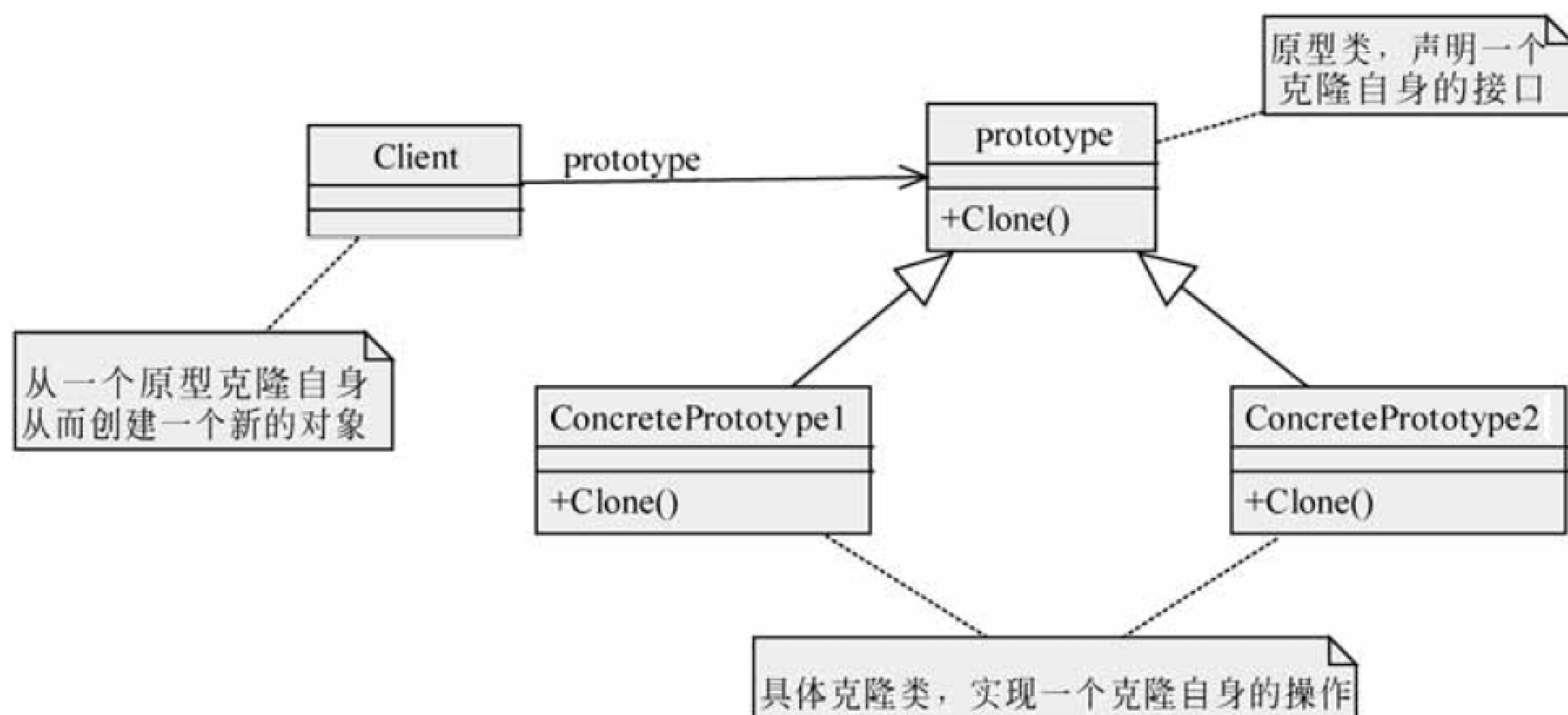
（1）向客户隐藏制造新实例的复杂性；

（2）提供让客户能够产生未知类型对象的选项；

（3）在某些环境下，复制对象比创建新对象更有效。

在一个复杂的类层次中，当系统必须从其中的许多类型创建新对象时，可以考虑原型模式。使用原型模式的缺点是：对象的复制有时相当复杂。

Prototype 模式的结构如下图所示。



题目利用原型模式来实现自动生成多份求职简历，而且每份简历的工作经历有所差别。题目中使用的是原型模型中的深克隆。题目中的类 Cloneable 对应上图中的类 Prototype，Resume 和 WorkExperience，分别是两个具体克隆类。由于 WorkExperience 是 Resume 的一部分，所以在 Resume 中有一个 WorkExperience 类型的成员对象。

下面来分析程序。

Cloneable 是 Java 中的一个接口，其中已经定义了 Clone()接口，可以直接在程序中使用。对于两个具体克隆类 Resume 和 WorkExperience 来说，只要实现 Cloneable 即可。因此第（1）、（3）空均填写 implements。

空（2）出现在类 WorkExperience 成员函数 Clone()的定义中。WorkExperience 作为具体克隆类，应实现 Cloneable 中的方法 Clone()。根据已给出的代码的提示，在 Clone()方法中，应再创建一个 WorkExperience 对象，而这个对象的名称就是 obj。因此（2）空处应填 WorkExperience obj = new WorkExperience()。

本题采用的是深克隆，除了对象本身被复制外，成员对象也被复制。work 是 Resume 中的成员对象，所以（3）空是对成员对象的克隆，因此应调用 work 对象的克隆操作，即(WorkExperience)work.Clone()。

第（4）空出现在 Resume 的 Clone()方法中，功能与第（2）空类似，完成 Resume 对象的克隆，所以第（4）空应填 new Resume(this.work)。

第（5）空是对原型模式的应用。用简历 a 克隆出简历 b，即调用 a 的克隆操作，所以第（5）空应填(Resume)a.Clone()。

参考答案

（1）implements

（2）WorkExperience obj = new WorkExperience()

（3）implements

（4）(WorkExperience)work.Clone()

（5）new Resume(this.work)

（6）(Resume)a.Clone()

第 19 章　2013 下半年软件设计师上午试题分析与解答

试题（1）

在程序执行过程中，Cache 与主存的地址映像由＿（1）＿。

（1）A. 硬件自动完成　　B. 程序员调度
　　C. 操作系统管理　　D. 程序员与操作系统协同完成

试题（1）分析

本题考查计算机系统基础知识。

Cache 的工作是建立在程序与数据访问的局部性原理上。经过对大量程序执行情况的结果分析：在一段较短的时间间隔内程序集中在某一较小的内存地址空间执行，这就是程序执行的局部性原理。同样，对数据的访问也存在局部性现象。

为了提高系统处理速度才将主存部分存储空间中的内容复制到工作速度更快的 Cache 中，同样为了提高速度的原因，Cache 系统都是由硬件实现的。

参考答案

（1）A

试题（2）

指令寄存器的位数取决于＿（2）＿。

（2）A. 存储器的容量　　B. 指令字长
　　C. 数据总线的宽度　　D. 地址总线的宽度

试题（2）分析

本题考查计算机系统基础知识。

指令寄存器是 CPU 中的关键寄存器，其内容为正在执行的指令，显然其位数取决于指令字长。

参考答案

（1）B

试题（3）

若计算机存储数据采用的是双符号位（00 表示正号、11 表示负号），两个符号相同的数相加时，如果运算结果的两个符号位经＿（3）＿运算得 1，则可断定这两个数相加的结果产生了溢出。

（3）A. 逻辑与　　B. 逻辑或　　C. 逻辑同或　　D. 逻辑异或

试题（3）分析

本题考查计算机系统基础知识。

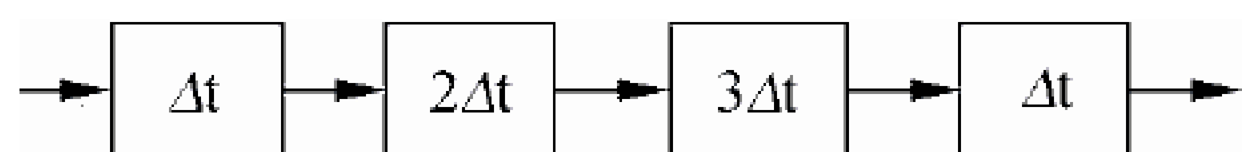
当表示数据时并规定了位数后，其能表示的数值范围就确定了，在两个数进行相加运算的结果超出了该范围后，就发生了溢出。在二进制情况下，溢出时符号位将变反，即两个正数相加，结果的符号位是负数，或者两个负数相加，结果的符号位是正数。采用两个符号位时，溢出发生后两个符号位就不一致了，这两位进行异或的结果一定为 1。

参考答案

（3）D

试题（4）

某指令流水线由 4 段组成，各段所需要的时间如下图所示。连续输入 8 条指令时的吞吐率（单位时间内流水线所完成的任务数或输出的结果数）为__（4）__。



（4）A. 8/56Δt　　B. 8/32Δt　　C. 8/28Δt　　D. 8/24Δt

试题（4）分析

本题考查计算机系统基础知识。

流水线的吞吐率指的是计算机中的流水线在特定的时间内可以处理的任务或输出数据的结果数量。流水线的吞吐率可以进一步分为最大吞吐率和实际吞吐率。该题目中要求解的是实际吞吐率，以流水方式执行 8 条指令的执行时间是 28Δt，因此吞吐率为 8/28Δt。

参考答案

（4）C

试题（5）

__（5）__不是 RISC 的特点。

（5）A. 指令种类丰富　　B. 高效的流水线操作
　　C. 寻址方式较少　　D. 硬布线控制

试题（5）分析

本题考查计算机系统基础知识。

RISC（Reduced Instruction Set Computer，精简指令集计算机）的主要特点是重叠寄存器窗口技术；优化编译技术。RISC 使用了大量的寄存器，如何合理分配寄存器、提高寄存器的使用效率及减少访存次数等，都应通过编译技术的优化来实现；超流水及超标量技术。为了进一步提高流水线速度而采用的技术；硬布线逻辑与微程序相结合在微程序技术中。

参考答案

（5）A

试题（6）

若某计算机字长为 32 位，内存容量为 2GB，按字编址，则可寻址范围为__（6）__。

（6）A. 1024M　　B. 1GB　　C. 512M　　D. 2GB

试题（6）分析

本题考查计算机系统基础知识。

内存容量 2GB=2*1024*1024*1024*8 位，按字编址时，存储单元的个数为 2*1024*1024*1024*8/32=512*1024*1024，即可寻址范围为 512MB。

参考答案

（6）C

试题（7）

下列网络攻击行为中，属于 DoS 攻击的是__（7）__。

（7）A. 特洛伊木马攻击　　B. SYN Flooding 攻击
C. 端口欺骗攻击　　D. IP 欺骗攻击

试题（7）分析

本试题考查网络安全相关知识。

特洛伊木马是附着在应用程序中或者单独存在的一些恶意程序，它可以利用网络远程控制网络另一端的安装有服务端程序的主机，实现对被植入了木马程序的计算机的控制，或者窃取被植入了木马程序的计算机上的机密资料。

拒绝服务攻击通过网络的内外部用户来发动攻击。内部用户可以通过长时间占用系统的内存、CPU 处理时间使其他用户不能及时得到这些资源，而引起拒绝服务攻击；外部黑客也可以通过占用网络连接使其他用户得不到网络服务。SYN Flooding 攻击以多个随机的源主机地址向目的路由器发送 SYN 包，在收到目的路由器的 SYN ACK 后并不回应，于是目的路由器就为这些源主机建立大量的连接队列，由于没有收到 ACK 一直维护着这些队列，造成了资源的大量消耗而不能向正常请求提供服务，甚至导致路由器崩溃。服务器要等待超时才能断开已分配的资源，所以 SYN Flooding 攻击是一种 DoS 攻击。

端口欺骗攻击是采用端口扫描找到系统漏洞从而实施攻击。

IP 欺骗攻击是产生的 IP 数据包为伪造的源 IP 地址，以便冒充其他系统或发件人的身份。

参考答案

（7）B

试题（8）分析

PKI 体制中，保证数字证书不被篡改的方法是__（8）__。

（8）A. 用 CA 的私钥对数字证书签名
B. 用 CA 的公钥对数字证书签名

C．用证书主人的私钥对数字证书签名

D．用证书主人的公钥对数字证书签名

试题（8）分析

本题考查 PKI 体制。

PKI 体制中，为保障数字证书不被篡改而且要发送到证书主人手中，需要用 CA 的私钥对数字证书签名，防伪造，不可抵赖。

参考答案

（8）A

试题（9）

下列算法中，不属于公开密钥加密算法的是__（9）__。

（9）A．ECC　　B．DSA　　C．RSA　　D．DES

试题（9）分析

本题考查加密算法的基础知识。

常用的加密算法依据所使用的秘钥数分为单钥和双钥加密体制，也称私钥和公钥加密算法。ECC、DSA 和 RSA 都属于公开密钥加密算法，DES 是典型的私钥加密体制。

参考答案

（9）D

试题（10）

矢量图是常用的图形图像表示形式，__（10）__是描述矢量图的基本组成单位。

（10）A．像素　　B．像素点　　C．图元　　D．二进制位

试题（10）分析

本题考查多媒体方面的基础知识。

矢量图形是用一系列计算机指令来描述和记录的一幅图的内容，即通过指令描述构成一幅图的所有直线、曲线、圆、圆弧、矩形等图元的位置、维数和形状，也可以用更为复杂的形式表示图像中的曲面、光照、材质等效果。矢量图法实质上是用数学的方式（算法和特征）来描述一幅图形图像，在处理图形图像时根据图元对应的数学表达式进行编辑和处理。在屏幕上显示一幅图形图像时，首先要解释这些指令，然后将描述图形图像的指令转换成屏幕上显示的形状和颜色。编辑矢量图的软件通常称为绘图软件，如适于绘制机械图、电路图的 AutoCAD 软件等。

参考答案

（10）C

试题（11）

视频信息是连续的图像序列，__（11）__是构成视频信息的基本单元。

（11）A．帧　　B．场　　C．幅　　D．像素

试题（11）分析

本题考查多媒体方面的基础知识。

视频信息是指活动的、连续的图像序列。一幅图像称为一帧，帧是构成视频信息的基本单元。

参考答案

（11）A

试题（12）

以下多媒体素材编辑软件中，__（12）__主要用于动画编辑和处理。

（12）A. WPS　　B. Xara3D　　C. PhotoShop　　D. Cool Edit Pro

试题（12）分析

本题考查多媒体编辑软件方面的知识。

多媒体编辑软件分为：文本工具、图形/图像工具、动画工具、视频工具、音频工具和播放工具。选项 A “WPS”属于文本工具类软件，主要用于文字编辑和处理；选项 B “Xara3D”属于动画工具类软件，主要用于动画编辑和处理；选项 C “PhotoShop”属于图形/图像工具类软件，主要用于显示图形/图像、图形/图像编辑、图像压缩、图像捕捉、图形/图像素材库；选项 D “Cool Edit Pro”属于音频工具类软件，主要用于音频播放、音频编辑、音频录制和声音素材库 4 个功能。

参考答案

（12）B

试题（13）

为说明某一问题，在学术论文中需要引用某些资料。以下叙述中，__（13）__是不正确的。

（13）A. 既可引用发表的作品，也可引用未发表的作品

B. 只能限于介绍、评论作品

C. 只要不构成自己作品的主要部分，可适当引用资料

D. 不必征得原作者的同意，不需要向他支付报酬

试题（13）分析

本题考查知识产权方面的基础知识。

选项 A “既可引用发表的作品，也可引用未发表的作品”的说法显然是错误的。因为，为说明某一问题，在学术论文中需要引用某些资料必须是已发表的作品，但只能限于介绍、评论作品，只要不构成自己作品的主要部分，可适当引用资料，而不必征得原作者的同意，不需要向他支付报酬。

参考答案

（13）A

试题（14）

以下作品中，不适用或不受著作权法保护的是__（14）__。

（14）A. 某教师在课堂上的讲课

B. 某作家的作品《红河谷》

C. 最高人民法院组织编写的《行政诉讼案例选编》

D. 国务院颁布的《计算机软件保护条例》

试题（14）分析

本题考查知识产权方面的基础知识。

选项 D“国务院颁布的《计算机软件保护条例》”的说法显然是错误的。因为，国务院颁布的《计算机软件保护条例》是国家为了管理需要制定的政策法规，故不适用著作权法保护。

参考答案

（14）D

试题（15）

以下关于数据流图中基本加工的叙述，不正确的是__（15）__。

（15）A. 对每一个基本加工，必须有一个加工规格说明

B. 加工规格说明必须描述把输入数据流变换为输出数据流的加工规则

C. 加工规格说明必须描述实现加工的具体流程

D. 决策表可以用来表示加工规格说明

试题（15）分析

本题考查结构化分析方法的基础知识。分层的数据流图是结构化分析方法的重要组成部分。对数据流图中的每个基本加工，需要有一个加工规格说明，描述把输入数据流变换为输出数据流的加工规则，但不需要描述实现加工的具体流程。可以用结构化语言、判定表和判定树来表达基本加工。

参考答案

（15）C

试题（16）

在划分模块时，一个模块的作用范围应该在其控制范围之内。若发现其作用范围不在其控制范围内，则__（16）__不是适当的处理方法。

（16）A. 将判定所在模块合并到父模块中，使判定处于较高层次

B. 将受判定影响的模块下移到控制范围内

C. 将判定上移到层次较高的位置

D. 将父模块下移，使该判定处于较高层次

试题（16）分析

本题考查软件设计的基础知识。

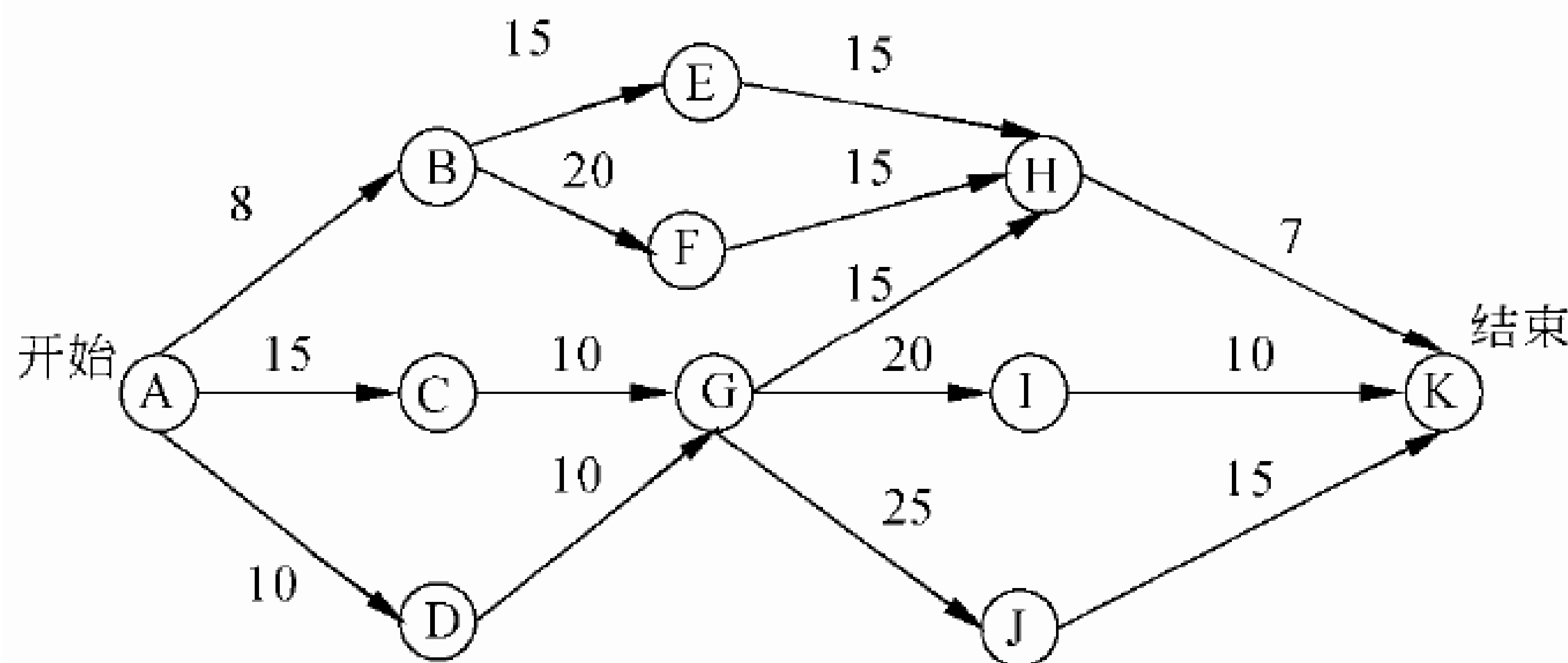
模块的控制范围包括模块本身及其所有的从属模块。模块的作用范围是指模块一个判定的作用范围，凡是受这个判定影响的所有模块都属于这个判定的作用范围。原则上一个模块的作用范围应该在其控制范围之内，若没有，则可以将判定所在模块合并到父模块中，使判定处于较高层次。

参考答案

（16）D

试题（17）、（18）

下图是一个软件项目的活动图，其中顶点表示项目里程碑，连接顶点的边表示包含的活动，则里程碑 __（17）__ 在关键路径上。若在实际项目进展中，活动 AD 在活动 AC 开始 3 天后才开始，而完成活动 DG 过程中，由于有临时事件发生，实际需要 15 天才能完成，则完成该项目的最短时间比原计划多了 __（18）__ 天。



（17）A. B　　B. C　　C. D　　D. I

（18）A. 8　　B. 3　　C. 5　　D. 6

试题（17）、（18）分析

本题考查软件项目管理的基础知识。

根据关键路径法，计算出关键路径为 A—C—G—I—K，关键路径长度为 65。因此里程碑 C 在关键路径上，而里程碑 B、D 和 I 不在关键路径上。

若完成活动 DG 需要 15 天，则相当于 A—D—G—I—K 也是一个关键路径，而且活动 AD 推迟了三天才能完成，此时，完成项目的最短时间应该是 68 天，比原来的最短时间 65 天多了 3 天。

参考答案

（17）B　　（18）B

试题（19）

针对“关键职员在项目未完成时就跳槽”的风险，最不合适的风险管理策略是 __（19）__。

（19）A. 对每一个关键性的技术人员，要培养后备人员

B. 建立项目组，以使大家都了解有关开发活动的信息

C. 临时招聘具有相关能力的新职员

D. 对所有工作组织细致的评审

试题（19）分析

本题考查软件项目管理的基础知识。

软件开发过程中不可避免会遇到风险，有效地管理软件风险对项目管理具有重要的意义。

对不同的风险采取不同的风险管理策略。如对关键职员在项目未完成时就跳槽的风险，可以通过培养后备人员、让项目组人员了解开发信息、评审开发工作等来降低风险。通过临时招聘新职员，即使新职员具有相关的能力，由于对项目的开发进展、团队组成等多种情况不了解，并不能很好地降低风险。

参考答案

（19）C

试题（20）

程序运行过程中常使用参数在函数（过程）间传递信息，引用调用传递的是实参的<u>（20）</u>。

（20）A. 地址　　B. 类型　　C. 名称　　D. 值

试题（20）分析

本题考查程序语言基础知识。

进行函数调用时，常需要在调用环境中的数据传递给被调用函数，作为输入参数由被调用函数处理，基本的调用方式为值调用（或传值调用）和引用调用。其中，值调用方式下是将实参的值单向地传递给被调用函数的形参，引用调用方式下通过将实参的地址传递给形参，在被调用函数中通过指针实现对实参变量数据的间接访问和修改，从而达到将修改后的值“传回来”的效果。

参考答案

（20）A

试题（21）

已知文法 G：S→A0|B1，A→S1|1，B→S0|0，其中 S 是开始符号。从 S 出发可以推导出<u>（21）</u>。

（21）A. 所有由 0 构成的字符串　　B. 所有由 1 构成的字符串

C. 某些 0 和 1 个数相等的字符串　　D. 所有 0 和 1 个数不同的字符串

试题（21）分析

本题考查程序语言基础知识。

用文法表示语言的语法规则时，推导是产生语言句子的基本方式。以题目中的文法为例，推导出 1010 的过程为 S=>A0=>S10=>A010=>1010，推导出 0110 的过程为 S=>A0=>S10=>B110=>0110，对于 0000、1111、1100、0011 等则推导不出。因为由 S 先

推导出A0后，再去推导A则必然产生一个与0相邻（在0的左边）的1，而由S先推导出B1，则下一步必然要推导出一个与1相邻（在1的左边）的0。这保证了当1出现时，马上就会出现0，或者反之，且0和1的距离很近。分析更多的例子发现，仅有“某些0和1个数相等的字符串”是正确的。

参考答案

（21）C

试题（22）

算术表达式a+(b-c)*d的后缀式是__（22）__（–、+、*表示算术的减、加、乘运算，运算符的优先级和结合性遵循惯例）。

（22）A. b c – d * a +　　B. a b c – d * +

C. a b + c – d *　　D. a b c d –* +

试题（22）分析

本题考查程序语言基础知识。

后缀式的特点是将运算符号写在运算数的后面。对于表达式，其计算次序是相减、相乘、相加，其后缀式为“abc-d*+”。

参考答案

（22）B

试题（23）

假设系统采用PV操作实现进程同步与互斥，若有n个进程共享一台扫描仪，那么当信号量S的值为-3时，表示系统中有__（23）__个进程等待使用扫描仪。

（23）A. 0　　B. n-3　　C. 3　　D. n

试题（23）分析

本题考查操作系统PV操作方面的基本知识。

系统采用PV操作实现进程的同步与互斥，当执行一次P操作表示申请一个资源，信号量S减1，如果S<0，其绝对值表示等待该资源的进程数。本题信号量S的值为-3，表示系统中有3个等扫描仪的进程。

参考答案

（23）C

试题（24）

假设段页式存储管理系统中的地址结构如下图所示，则系统中__（24）__。

31　　24	23　　12	11　　0
段　号	页　号	页内地址

（24）A. 页的大小为4K，每个段的大小均为4096个页，最多可有256个段

B. 页的大小为 4K，每个段最大允许有 4096 个页，最多可有 256 个段

C. 页的大小为 8K，每个段的大小均为 2048 个页，最多可有 128 个段

D. 页的大小为 8K，每个段最大允许有 2048 个页，最多可有 128 个段

试题（24）分析

本题考查操作系统页式存储管理方面的基础知识。从图中可见，页内地址的长度是 12 位，$2^{12}=4096$，即 4K；页号部分的地址长度是 12 位，每个段最大允许有 4096 个页；段号部分的地址长度是 8 位，$2^{8}=256$，最多可有 256 个段。

参考答案

（24）B

试题（25）、（26）

某文件管理系统采用位示图（bitmap）记录磁盘的使用情况。如果系统的字长为 32 位，磁盘物理块的大小为 4MB，物理块依次编号为：0、1、2、…，位示图字依次编号为：0、1、2、…，那么 16385 号物理块的使用情况在位示图中的第__（25）__个字中描述；如果磁盘的容量为 1000GB，那么位示图需要__（26）__个字来表示。

（25）A. 128　　B. 256　　C. 512　　D. 1024

（26）A. 1200　　B. 3200　　C. 6400　　D. 8000

试题（25）、（26）分析

本题考查操作系统文件管理方面的基本知识。

文件管理系统是在外存上建立一张位示图（bitmap），记录文件存储器的使用情况。每一位对应文件存储器上的一个物理块，取值 0 和 1 分别表示空闲和占用，如下图所示。

第 0 字	1	0	1	0	0	…	1	1
第 1 字	0	1	1	1	0	…	0	1
第 2 字	1	1	1	1	0	…	1	0
⋮			⋮					
第 n−1 字	0	0	0	1	1	…	0	0

位示图例

由于系统中字长为 32 位，所以每个字可以表示 32 个物理块的使用情况。又因为文件存储器上的物理块依次编号为：0、1、2、…，位示图表示物理块的情况如下，从下图可见，16385 号物理块应该在位示图的第 512 个字中描述。

又因为磁盘物理块的大小为 4MB，1GB=1024M=256 个物理块，需要 8 个字表示，故磁盘的容量为 1000GB，那么位示图需要 1000×8=8000 个字表示。

第 0 字	1	0	1	0	0	…	1	1	表示 0,1,…,31 物理块使用情况
第 1 字	0	1	1	1	0	…	0	1	表示 32,33,…,63 物理块使用情况
第 2 字	1	1	1	1	0	…	1	0	表示 64,65,…,95 物理块使用情况
⋮	⋮								⋮
第 512 字	0	0	0	1	1	…	0	0	表示 16384, 16385,…,16415 物理块使用情况

位示图例

参考答案

（25）C　（26）D

试题（27）、（28）

假设系统中有三类互斥资源 R1、R2 和 R3，可用资源数分别为 10、5 和 3。在 T_0 时刻系统中有 P1、P2、P3、P4 和 P5 五个进程，这些进程对资源的最大需求量和已分配资源数如下表所示，此时系统剩余的可用资源数分别为__（27）__。如果进程按__（28）__序列执行，那么系统状态是安全的。

资源 进程	最大需求量			已分配资源数		
	R1	R2	R3	R1	R2	R3
P1	5	3	1	1	1	1
P2	3	2	0	2	1	0
P3	6	1	1	3	1	0
P4	3	3	2	1	1	1
P5	2	1	1	1	1	0

（27）A. 1、1 和 0　　B. 1、1 和 1　　C. 2、1 和 0　　D. 2、0 和 1

（28）A. P1→P2→P4→P5→P3　　B. P5→P2→P4→P3→P1

C. P4→P2→P1→P5→P3　　D. P5→P1→P4→P2→P3

试题（27）、（28）分析

初始时系统的可用资源数分别为 10、5 和 3。在 T_0 时刻已分配资源数分别为 8、5 和 2，因此系统剩余的可用资源数分别为 2、0 和 1。

安全状态是指系统能按某种进程顺序（P1,P2,⋯,Pn），为每个进程 Pi 分配其所需的资源，直到满足每个进程对资源的最大需求，使每个进程都可以顺利完成。如果无法找到这样的一个安全序列，则称系统处于不安全状态。

本题进程的执行序列已经给出，我们只需将四个选项按其顺序执行一遍，便可以判断出现死锁的三个序列。

资源 / 进程	最大需求量			已分配资源数			尚需资源数		
	R1	R2	R3	R1	R2	R3	R1	R2	R3
P1	5	3	1	1	1	1	4	2	0
P2	3	2	1	2	1	0	1	1	1
P3	6	1	1	3	1	0	3	0	1
P4	3	3	2	1	1	1	2	2	1
P5	2	1	1	1	1	0	1	0	1

选项 A P1→P2→P4→P5→P3 是不安全的序列。因为在该序列中，进程 P1 先运行，P1 尚需资源数为（4，2，0），假设将资源 R1 分配 2 台给进程 P1，则系统剩余的可用资源数为（0，0，1），将导致系统所有的进程都不能标上能完成标志“True”，故选项 A 是不安全的序列。

选项 B P5→P2→P4→P3→P1 是安全的序列。因为所有的进程都能标上能完成标志“True”，如下表所示。

资源 / 进程	可用资源数 R1 R2 R3	已分配资源数 R1 R2 R3	尚需资源数 R1 R2 R3	可用+已分 R1 R2 R3	能否完成标志
P5	2　0　1	1　1　0	1　0　1	3　1　1	True
P2	3　1　1	2　1　0	1　1　1	5　2　1	True
P4	5　2　1	1　1　1	2　2　1	6　3　2	True
P3	6　3　2	3　1　0	3　0　1	9　4　2	True
P1	9　4　2	1　1　1	4　2　0	10　5　3	True

对序列 P5→P2→P4→P3→P1 的具体分析如下：

① 进程 P5 运行，系统剩余的可用资源数为（2，0，1），P5 尚需资源数为（1，0，1），系统可进行分配，故进程 P5 能标上能完成标志“True”，释放 P5 占有的资源数（1，1，0），系统可用资源数为（3，1，1）。

② 进程 P2 运行，系统剩余的可用资源数为（3，1，1），P2 尚需资源数为（1，1，1），系统可进行分配，故进程 P2 能标上能完成标志“True”，释放 P2 占有的资源数（2，1，0），系统可用资源数为（5，2，1）。

③ 进程 P4 运行，系统剩余的可用资源数为（5，2，1），P4 尚需资源数为（2，2，1），系统可进行分配，故进程 P4 能标上能完成标志“True”，释放 P4 占有的资源数（1，1，1），系统可用资源数为（6，3，2）。

④ 进程 P3 运行，系统剩余的可用资源数为（6，3，2），P3 尚需资源数为（3，0，1），系统可进行分配，故进程 P3 能标上能完成标志“True”，释放 P3 占有的资源数（3，1，0），系统可用资源数为（9，4，2）。

⑤ 进程 P1 运行，系统剩余的可用资源数为（9，4，2），P1 尚需资源数为（4，2，0），系统可进行分配，故进程 P1 能标上能完成标志“True”，释放 P1 占有的资源数（1，1，1），

系统可用资源数为（10，5，3）。

P4→P2→P1→P5→P3 是不安全的序列。因为在该序列中，进程 P4 先运行，P4 尚需资源数为（2，2，1），假设将资源 R1 分配 2 台给进程 P4，则系统剩余的可用资源数为（0，0，1），将导致系统所有的进程都不能标完成标志"True"。

P5→P1→P4→P2→P3 是不安全的序列。因为在该序列中，进程 P5 先运行，系统剩余的可用资源数为（2，0，1），P5 尚需资源数为（1，0，1），系统可进行分配，故进程 P5 能标上能完成标志"True"，释放 P5 占有的资源数（1，1，0），系统可用资源数为（3，1，1）。进程 P1 运行，P1 尚需资源数为（4，2，0），假设将资源 R1 分配 3 台给进程 P1，则系统剩余的可用资源数为（0，1，1），将导致系统中的进程 P1、P2、P3 和 P4 都不能标上能完成标志"True"。

参考答案

（27）D （28）B

试题（29）

__（29）__开发过程模型最不适用于开发初期对软件需求缺乏准确全面认识的情况。

（29）A. 瀑布 B. 演化 C. 螺旋 D. 增量

试题（29）分析

本题考查软件过程模型的基础知识。瀑布模型将软件生存周期各个活动规定为线性顺序连接的若干阶段的模型，规定了由前至后，相互衔接的固定次序，如同瀑布流水，逐级下落。这种方法是一种理想的现象开发模式，缺乏灵活性，特别是无法解决软件需求不明确或不准确的问题。演化模型从初始的原型逐步演化成最终软件产品，特别适用于对软件需求缺乏准确认识的情况。螺旋将瀑布模型与快速原型模型结合起来，并且加入两种模型均忽略了的风险分析，适用于复杂的大型软件。增量开发是把软件产品作为一系列的增量构件来设计、编码、集成和测试，可以在增量开发过程中逐步理解需求。

参考答案

（29）A

试题（30）

__（30）__不是增量式开发的优势。

（30）A. 软件可以快速地交付

B. 早期的增量作为原型，从而可以加强对系统后续开发需求的理解

C. 具有最高优先级的功能首先交付，随着后续的增量不断加入，这就使得更重要的功能得到更多的测试

D. 很容易将客户需求划分为多个增量

试题（30）分析

本题考查过程模型的基础知识。增量开发是把软件产品作为一系列的增量构件来设计、编码、集成和测试。每个构件由多个相互作用的模块构成，并且能够完成特定的功

能。其优点包括：能在较短时间内向用户提交可完成一些有用的工作产品；逐步增加产品的功能可以使用户有较充裕的时间学习和适应新产品；项目失败的风险较低；优先级高的服务首先交付，使得最重要的系统服务将接受最多的测试。

参考答案

（30）D

试题（31）

在对程序质量进行评审时，模块结构是一个重要的评审项，评审内容中不包括（31）。

（31）A. 数据结构　　B. 数据流结构
C. 控制流结构　　D. 模块结构与功能结构之间的对应关系

试题（31）分析

本题考查软件质量的基础知识。程序质量评审通常是从开发者的角度进行，与开发技术直接相关，考虑软件本身的结构、与运行环境的接口以及变更带来的影响等。其中，软件结构包括功能结构、功能的通用性、模块的层次性、模块结构和处理过程的结构，而模块结构包括控制流结构、数据流结构、模块结构与功能结构之间的对应关系。

参考答案

（31）A

试题（32）

SEI 能力成熟度模型（SEI CMM）把软件开发企业分为 5 个成熟度级别，其中（32）重点关注产品和过程质量。

（32）A. 级别 2：重复级　　B. 级别 3：确定级
C. 级别 4：管理级　　D. 级别 5：优化级

试题（32）分析

本题考查软件过程和软件过程改进的基础知识。CMM 是指软件开发能力成熟度模型，该模型给出了从混乱的个别的过程达到成熟的规范化过程的一个框架，分成 5 个等级，从 1 级到 5 级成熟度逐步提高。级别 1 为初始级，特点是混乱和不可预测；级别 2 为重复级级别，特点是项目得到管理监控和跟踪，有稳定的策划和产品基线；级别 3 为确定级级别，通过软件过程的定义和制度化确保对产品质量的控制；级别 4 为管理级级别，特点是产品质量得到策划，软件过程基于度量的跟踪；级别 5 为优化级，特点是持续的过程能力改进。

参考答案

（32）C

试题（33）

系统可维护性的评价指标不包括（33）。

（33）A. 可理解性　　B. 可测试性

C. 可移植性　　　　D. 可修改性

试题（33）分析

本题考查软件质量的基础知识。软件的可维护性是指纠正软件系统出现的错误和缺陷，以及为满足新的要求进行修改、扩充或压缩的容易程度，是软件开发阶段各个时期的关键目标。其中，可理解性、可测试性和可修改性是衡量可维护性的重要指标。

参考答案

（33）C

试题（34）

逆向工程从源代码或目标代码中提取设计信息，通常在原软件生命周期的（34）阶段进行。

（34）A. 需求分析　　　　B. 软件设计

C. 软件实现　　　　D. 软件维护

试题（34）分析

本题考查软件工程的基础知识。逆向工程从详细的源代码实现中抽取抽象规格说明，一般来说是在原软件交付用户使用之后进行的，即在原软件的维护阶段进行。

参考答案

（34）D

试题（35）

一个程序根据输入的年份和月份计算该年中该月的天数，输入参数包括年份（正整数）、月份（用 1~12 表示）。若用等价类划分测试方法进行测试，则（35）不是一个合适的测试用例（分号后表示测试的输出）。

（35）A. (2013,1; 31)　　　　B. (0,1; '错误')

C. (0,13; '错误')　　　　D. (2000,-1; '错误')

试题（35）分析

本题考查软件测试的基础知识。常用的测试技术包括白盒测试和黑盒测试。白盒测试是利用程序内部的逻辑结构及有关信息，设计或选择测试用例，对程序所在逻辑路径进行测试，又称为结构测试或逻辑驱动测试。黑盒测试根据程序的需求规格说明书，检查程序的功能是否符合它的功能说明。

等价类划分是一类黑盒测试技术，该方法把输入数据分为若干个等价类，包括有效的和无效的等价类。基于等价类设计测试用例时，每个测试用例至多覆盖一个无效等价类，选项 C 包含两个无效等价类，故不是一个好的测试用例。

参考答案

（35）C

试题（36）

（36）不是单元测试主要检查的内容。

（36）A. 模块接口　　B. 局部数据结构
C. 全局数据结构　　D. 重要的执行路径

试题（36）分析

本题考查软件测试的基础知识。单元测试又称为模块测试，是针对软件设计的最小单元（程序模块），进行正确性检验的测试。其目的在于发现个模块内不可能存在的各种问题和错误。单元测试需要从程序的内部结构出发设计测试用例。模块可以单独进行单元测试。单元测试测试以下几个方面：模块接口、局部数据结构、执行路径、错误处理和边界。

参考答案

（36）C

试题（37）

在领域类模型中不包含＿（37）＿。

（37）A. 属性　　B. 操作　　C. 关联　　D. 领域对象

试题（37）分析

本题考查面向对象的基本知识。

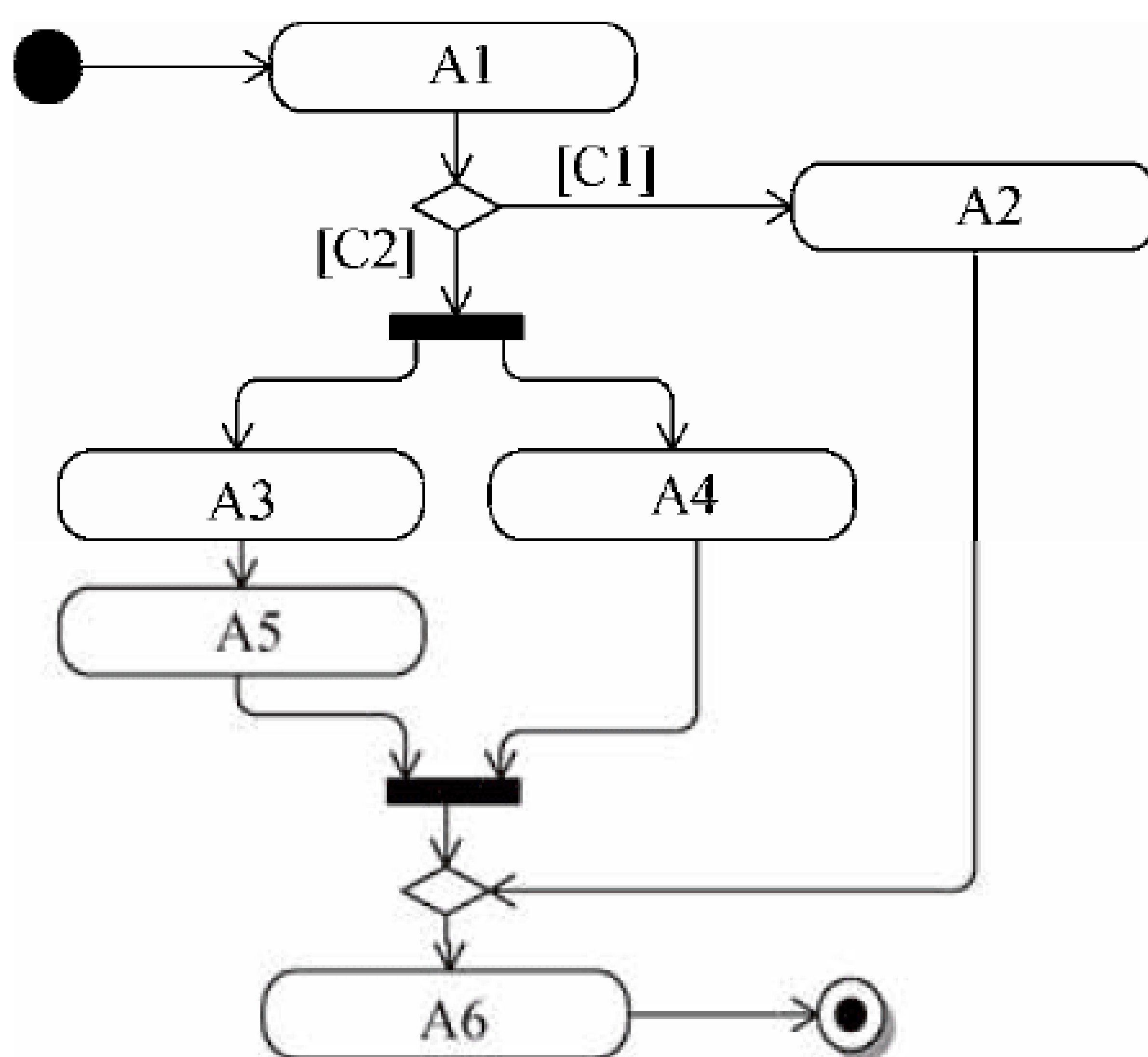
定义领域模型是面向对象分析的关键步骤之一。领域模型是从按对象分类的角度来创建对象领域的描述，包括定义概念、属性和重要的关联，其结果用一组显示领域概念和对象的图形——类图来组织，图中还包括多重性、关联关系、泛化/特化关系以及聚合关系等。

参考答案

（37）D

试题（38）

在执行如下所示的 UML 活动图时，能同时运行的最大线程数为＿（38）＿。



（38）A. 4　　B. 3　　C. 2　　D. 1

试题（38）分析

本题考查统一建模语言（UML）的基本知识。

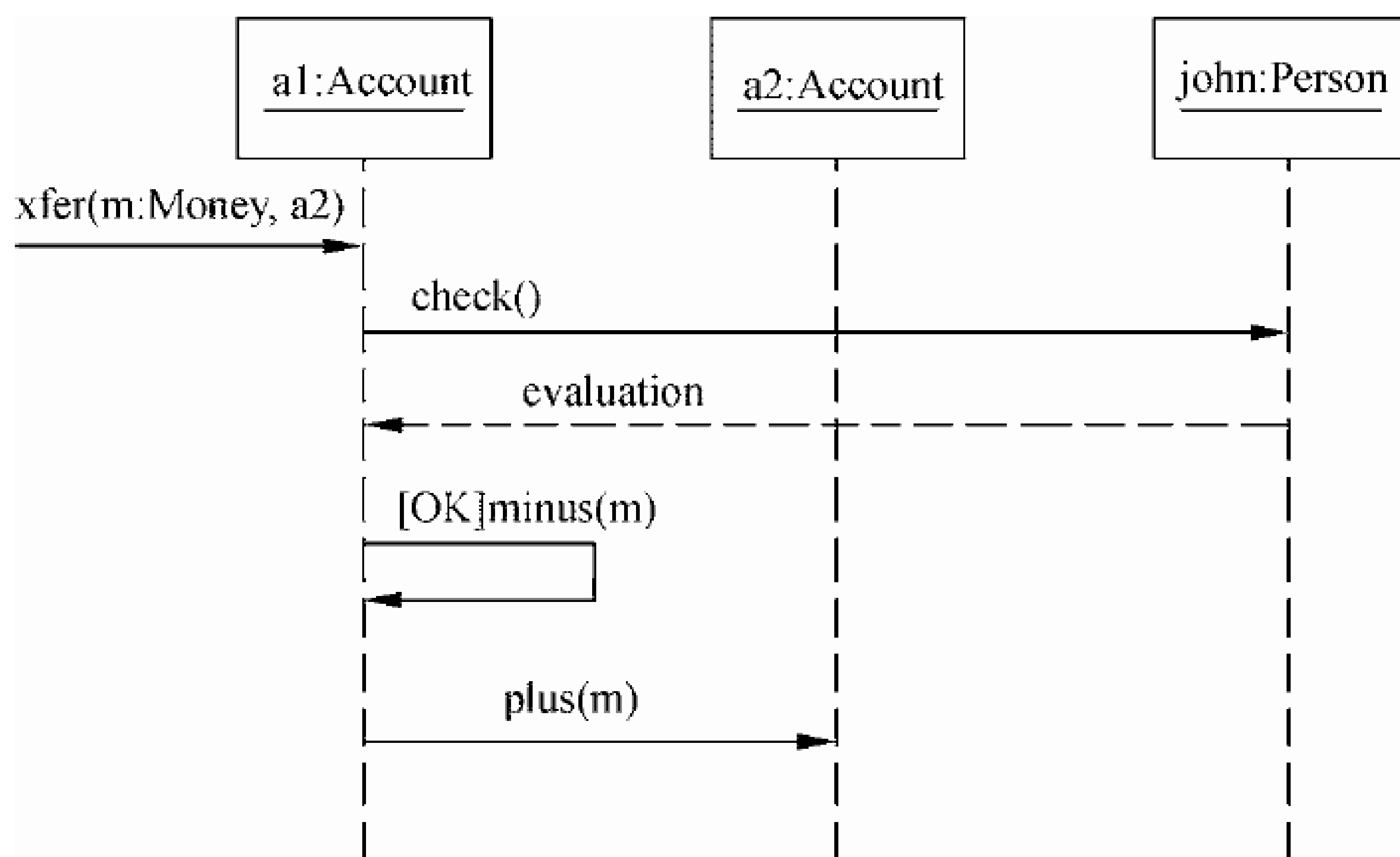
UML 活动图用于构建系统的活动。建模用例执行过程中对象如何通过消息相互交互，将系统作为一个整体或者几个子系统进行考虑。对象在运行时可能会存在两个或多个并发运行的控制流，为了对并发控制流进行建模，UML 中引入同步的概念，用同步棒——黑色粗线条表示并发分支与汇合。

参考答案

（38）C

试题（39）、（40）

下图所示的 UML 序列图中，（39）表示返回消息，Account 应该实现的方法有（40）。



（39）A. xfer　　B. check　　C. evaluation　　D. minus

（40）A. xfer()

B. xfer()、plus()和 minus()

C. check()、plus()和 minus()

D. xfer()、evaluation()、plus()和 minus()

试题（39）、（40）分析

本题考查统一建模语言（UML）的基本知识。

UML 序列图（Sequence Diagram）以二维图的形式显示对象之间交互的图，纵轴自上而下表示时间，横轴表示要交互的对象，主要体现对象间消息传递的时间顺序，强调参与交互的对象及其间消息交互的时序。序列图中包括的建模元素主要有：活动者（Actor）、对象（Object）、生命线（Lifeline）、控制焦点（Focus of control）和消息（Message）等。其中对象名标有下划线；生命线表示为虚线，沿竖线向下延伸；消息在序列图中标记为箭头；控制焦点由薄矩形表示。

消息是从一个对象的生命线到了一个对象生命线的箭头，用从上而下的时间顺序来安排。一般分为同步消息（ ⟶ ），异步消息（ ⟶ ）和返回消息（ ←-- ）。本题图中 evaluation 为返回消息，其他为同步消息。a1 和 a2 均为 Account 对象，所以 Account 应该实现了 xfer()、minus()和 plus()方法，Person 应该实现 check()方法。

参考答案

（39）C　　（40）B

试题（41）～（43）

在面向对象技术中，__（41）__定义了超类和子类之间的关系，子类中以更具体的方式实现从父类继承来的方法称为__（42）__，不同类的对象通过__（43）__相互通信。

（41）A. 覆盖　　B. 继承　　C. 消息　　D. 多态

（42）A. 覆盖　　B. 继承　　C. 消息　　D. 多态

（43）A. 覆盖　　B. 继承　　C. 消息　　D. 多态

试题（41）～（43）分析

本题考查面向对象技术的基础知识。

在面向对象技术中，继承关系是一种模仿现实世界中继承关系的一种类之间的关系，是超类（父类）和子类之间共享数据和方法的机制。在定义和实现一个类的时候，可以在一个已经存在的类的基础上来进行，子类可以继承其父类中的属性和操作作为自己的内容而不必自己定义，也可以用更具体地方式实现从父类继承来的方法，称为覆盖。不同的对象收到同一消息可以进行不同的响应，产生完全不同的结果，用户可以发送一个通用的消息，而实现细节则由接收对象自行决定，使得同一个消息就可以调用不同的方法，即一个对象具有多种形态，称为多态。不同类的对象通过消息相互通信。

参考答案

（41）B　　（42）A　　（43）C

试题（44）

__（44）__设计模式定义一系列算法，把它们一个个封装起来，并且使它们可相互替换。这一模式使得算法可独立于它的客户而变化。

（44）A. 策略（Strategy）　　B. 抽象工厂（Abstract Factory）

C. 观察者（Visitor）　　D. 状态（State）

试题（44）分析

本题考查设计模式的基本概念。

策略（Strategy）设计模式定义一系列算法，把它们一个个封装起来，并且使它们可相互替换。这一模式使得算法可独立于它的客户而变化。抽象工厂（Abstract Factory）模式提供一个创建一系列相关或相互依赖对象的接口，而无须指定他们具体的类。观察者（Observer）模式定义对象间的一种一对多的依赖关系，当一个对象的状态发生改变时，所有依赖于它的对象都得到通知并被自动更新。状态（State）模式是使得一个对象

在其内部状态改变时通过调用另一个类中的方法改变其行为，使这个对象看起来如同修改了它的类。

参考答案

（44）A

试题（45）

在发布-订阅（Publish-Subscribe）消息模型中，订阅者订阅一个主题后，当该主题有新消息到达时，所有订阅者都会收到通知。__（45）__设计模式最适合这一模型。

（45）A. 适配器（Adapter） B. 通知（Notifier）
C. 状态（State） D. 观察者（Observer）

试题（45）分析

本题考查设计模式的基本概念。

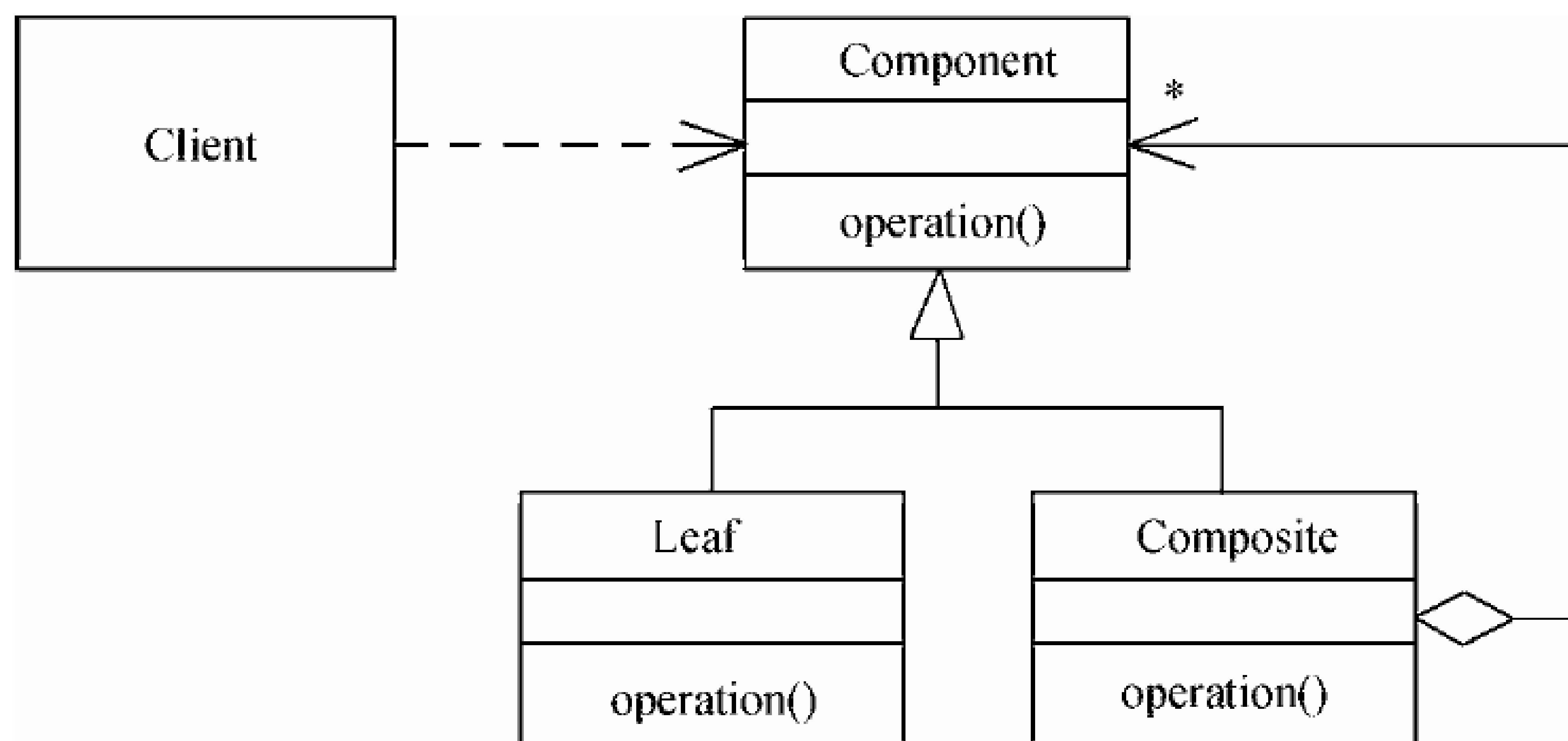
适配器（Adapter）模式将一个类的接口转换成客户希望的另外一个接口，使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。观察者（Observer）模式定义对象间的一种一对多的依赖关系，当一个对象的状态发生改变时，所有依赖于它的对象都得到通知并被自动更新，其别名为发布-订阅（Publish-Subscribe）模式。状态（State）模式是使得一个对象在其内部状态改变时通过调用另一个类中的方法改变其行为，使这个对象看起来如同修改了它的类。

参考答案

（44）D

试题（46）、（47）

下图所示为__（46）__设计模式，适用于：__（47）__。



（46）A. 组件（Component） B. 适配器（Adapter）
C. 组合（Composite） D. 装饰器（Decorator）

（47）A. 表示对象的部分一整体层次结构
B. 不希望在抽象和它的实现部分之间有一个固定的绑定关系

C. 在不影响其他对象的情况下，以动态、透明的方式给单个对象添加职责

D. 使所有接口不兼容类可以一起工作

试题（46）、（47）分析

本题考查设计模式的基本概念。每种设计模式都有特定的意图，描述一个在我们周围不断重复发生的问题，以及该问题的解决方案的核心，使该方案能够重用而不必做重复劳动。

适配器（Adapter）模式将一个类的接口转换成客户希望的另外一个接口，使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。

组合（Composite）模式将对象组合成树形结构以表示“部分-整体”的层次结构，使得用户对单个对象和组合对象的使用具有一致性。组件 Component 为组合的对象声明接口，通常定义父组件引用，Leaf 和 Composite 类可以继承这个引用以及管理这个应用的那些操作。

装饰器（Decorator）模式描述了以透明围栏来支持修饰的类和对象的关系，动态地给一个对象添加一些额外的职责，从增加功能的角度来看，装饰器模式相比生成子类更加灵活。

参考答案

（46）C　　（47）A

试题（48）

将高级语言程序翻译为机器语言程序的过程中，常引入中间代码，其好处是__（48）__。

（48）A. 有利于进行反编译处理　　B. 有利于进行与机器无关的优化处理

C. 尽早发现语法错误　　D. 可以简化语法和语义分析

试题（48）分析

本题考查程序语言基础知识。

“中间代码”是一种简单且含义明确的记号系统，可以有若干种形式，它们的共同特征是与具体的机器无关，此时所作的优化一般建立在对程序的控制流和数据流分析的基础之上，与具体的机器无关。

参考答案

（48）B

试题（49）

对高级语言源程序进行编译的过程中，有穷自动机（NFA 或 DFA）是进行__（49）__的适当工具。

（49）A. 词法分析　　B. 语法分析　　C. 语义分析　　D. 出错处理

试题（49）分析

本题考查程序语言基础知识。

语言中具有独立含义的最小语法单位是符号（单词），如标识符、无符号常数与界

限符等。词法分析的任务是把构成源程序的字符串转换成单词符号序列。

有限自动机是一种识别装置的抽象概念，它能准确地识别正规集。有限自动机分为两类：确定的有限自动机（DFA）和不确定的有限自动机（NFA）。

参考答案

（49）A

试题（50）

弱类型语言（动态类型语言）是指不需要进行变量/对象类型声明的语言。__（50）__属于弱类型语言。

（50）A. Java B. C/C++ C. Python D. C#

试题（50）分析

本题考查程序语言基础知识。

弱/强类型指的是语言类型系统的类型检查的严格程度，动态类型和静态类型则指变量与类型的绑定方法。

静态类型指编译器在编译源程序期间执行类型检查，动态类型指编译器（虚拟机）在程序运行时执行类型检查。简单地说，在声明了一个变量之后，不能改变其类型的语言，是静态语言；能够随时改变其类型的语言，是动态语言。

弱类型相对于强类型来说类型检查更不严格，比如说允许变量类型的隐式转换，允许强制类型转换等等。

参考答案

（50）C

试题（51）～（54）

若有关系 R（A，B，C，D，E）和 S（B，C，F，G），则 R 与 S 自然联结运算后的属性列有__（51）__个，与表达式 $\pi_{1,3,6,7}(\sigma_{3<6}(R \bowtie S))$ 等价的 SQL 语句如下：

SELECT __（52）__ FROM __（53）__ WHERE __（54）__;

（51）A. 5 B. 6 C. 7 D. 9

（52）A. A, R.C, F, G B. A, C, S.B, S.F

C. A, C, S.B, S.C D. R.A, R.C, S.B, S.C

（53）A. R B. S C. RS D. R, S

（54）A. R.B = S.B AND R.C = S.C AND R.C < S.B

B. R.B = S.B AND R.C = S.C AND R.C < S.F

C. R.B = S.B OR R.C = S.C OR R.C < S.B

D. R.B = S.B OR R.C = S.C OR R.C < S.F

试题（51）～（54）分析

本题考查关系代数运算与 SQL 查询方面的基础知识。

在 $\pi_{1,3,6,7}(\sigma_{3<6}(R \bowtie S))$ 中，自然联结 R ⋈ S 运算后去掉右边重复的属性列名 S.B、S.C 后为：R.A、R.B、R.C、R.D、R.E、S.F 和 S.G，因此空（51）的正确答案为 7。$\pi_{1,3,6,7}(\sigma_{3<6}(R \bowtie S))$ 的含义是从 R ⋈ S 结果集中选取 R.C < S.F 的元组，再进行 R.A、R.C、S.F 和 S.G 投影，因此，空（52）的正确答案为选项 A。显然，空（53）的答案为 R, S。

空（54）的正确答案为选项 B。因为，自然联结 R ⋈ S 需要用条件“WHERE R.B =S.B AND R.C =S.C”来限定，选取运算 $\sigma_{3<6}$ 需要用条件“WHERE　R.C<S.F”来限定。

参考答案

（51）C　（52）A　（53）D　（54）B

试题（55）

在分布式数据库系统中，__（55）__是指用户无须知道数据存放的物理位置。

（55）A. 分片透明　　B. 复制透明　　C. 逻辑透明　　D. 位置透明

试题（55）分析

本题考查分布式数据库基本概念。分片透明是指用户或应用程序不需要知道逻辑上访问的表具体是怎么分块存储的。复制透明是指采用复制技术的分布方法，用户不需要知道数据是复制到哪些节点，如何复制的。位置透明是指用户无须知道数据存放的物理位置，逻辑透明局部数据模型透明，是指用户或应用程序无须知道局部场地使用的是哪种数据模型。

参考答案

（55）D

试题（56）

计算机系统的软硬件故障可能会造成数据库中的数据被破坏。为了防止这一问题，通常需要__（56）__，以便发生故障时恢复数据库。

（56）A. 定期安装 DBMS 和应用程序
B. 定期安装应用程序，并将数据库做镜像
C. 定期安装 DBMS，并将数据库作备份
D. 定期将数据库作备份；在进行事务处理时，需要将数据更新写入日志文件

试题（56）分析

本题考查关系数据库事务处理方面的基础知识。

为了保证数据库中数据的安全可靠和正确有效，数据库管理系统（DBMS）提供数据库恢复、并发控制、数据完整性保护与数据安全性保护等功能。数据库在运行过程中由于软硬件故障可能造成数据被破坏，数据库恢复就是在尽可能短的时间内，把数据库恢复到故障发生前的状态。具体的实现方法有多种，如：定期将数据库作备份；在进行事务处理时，对数据更新（插入、删除、修改）的全部有关内容写入日志文件；当系统正常运行时，按一定的时间间隔，设立检查点文件，把内存缓冲区内容还未写入到磁盘

中去的有关状态记录到检查点文件中；当发生故障时，根据现场数据内容、日志文件的故障前映像和检查点文件来恢复系统的状态。

参考答案

（56）D

试题（57）

以下关于线性表存储结构的叙述，正确的是__（57）__。

（57）A. 线性表采用顺序存储结构时，访问表中任意一个指定序号元素的时间复杂度为常量级

B. 线性表采用顺序存储结构时，在表中任意位置插入新元素的运算时间复杂度为常量级

C. 线性表采用链式存储结构时，访问表中任意一个指定序号元素的时间复杂度为常量级

D. 线性表采用链式存储结构时，在表中任意位置插入新元素的运算时间复杂度为常量级

试题（57）分析

本题考查数据结构基础知识。

线性表进行顺序存储时，逻辑上相邻的元素，其物理位置也相邻，因此在已知第一个元素存储位置和元素序号的情况下，可计算出表中任意指定序号元素的存储位置，即按照序号访问元素时随机的，该运算的时间复杂度为 O(1)，也就是常量级。而插入元素时就需要移动一些元素了，在最坏情况下要移动表中的所有元素，因此该运算的时间复杂度为 O(n)，其中 n 为线性表的长度。

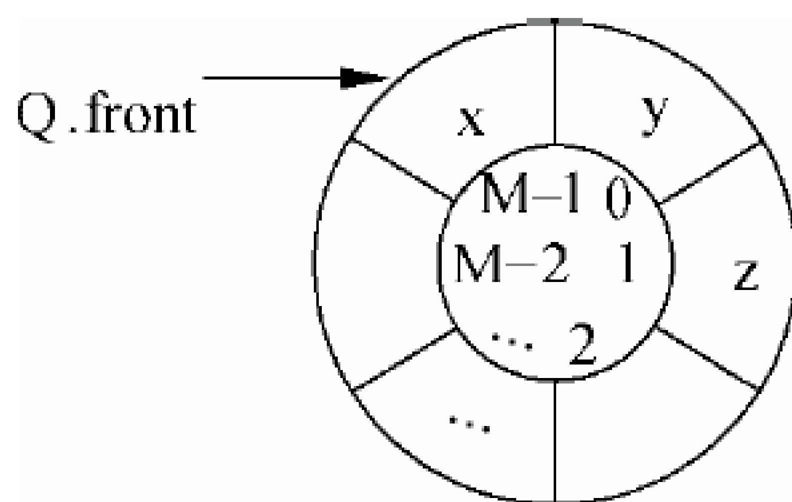
线性表进行链式存储时，逻辑上相邻的元素，其物理位置不要求相邻，因此需要额外的存储空间表示元素之间的顺序关系。在链表上查找元素和插入元素的运算时间复杂度都为 O(n)。

参考答案

（57）A

试题（58）

设循环队列 Q 的定义中有 front 和 size 两个域变量，其中 front 表示队头元素的指针，size 表示队列的长度，如下图所示（队列长度为 3，队头元素为 x、队尾元素为 z）。设队列的存储空间容量为 M，则队尾元素的指针为__（58）__。



（58）A.　(Q.front+Q.size-1)　　　B.　(Q. front +Q.size-1+M) % M

　　C.　(Q. front-Q.size)　　　D.　(Q. front-Q.size+M) % M

试题（58）分析

本题考查数据结构基础知识。

根据题目中所给的示意图，Q.front 为队头元素的指针，该指针加 1 后得到队列中的第 2 个元素（即 y）的指针，由于队列中存储位置编号是在 0~M-1 之间循环的，队头指针加上 1 个增量后可能会超出该范围，应该用整除取余运算恢复一下，因此由 Q.front 可以算出队列尾部元素的指针为(Q. front +Q.size-1+M) % M。

参考答案

（58）B

试题（59）

在一个有向图 G 的拓扑序列中，顶点 v_i 排列在 v_j 之前，说明图 G 中__（59）__。

（59）A. 一定存在弧 $< v_i, v_j >$　　　B. 一定存在弧

C. 可能存在 v_i 到 v_j 的路径，而不可能存在 v_j 到 v_i 的路径

D. 可能存在 v_j 到 v_i 的路径，而不可能存在 v_i 到 v_j 的路径

试题（59）分析

本题考查数据结构基础知识。

对一个有向图 G 进行拓扑排序的方法如下。

① G 中选择一个入度为 0（没有前驱）的顶点且输出它；

② 从网中删除该顶点及其与该顶点有关的所有弧；

③ 重复上述两步，直至网中不存在入度为 0 的顶点为止。

显然，若存在弧 $< v_i, v_j >$，则 v_j 的入度就不为 0，而要删除该弧，则 v_i 的入度应为 0，因此在拓扑序列中，v_i 必然在 v_j 之前。另外，进行拓扑排序时，可能存在 v_i 和 v_j 的入度同时为 0 的情形，此时，在第①步可先输出 v_i，后输出 v_j。因此在拓扑序列中，顶点 v_i 排列在 v_j 之前，不一定存在弧 $< v_i, v_j >$，一定不存在弧 $< v_j, v_i >$，也一定不存在 v_j 到 v_i 的路径，而可能存在 v_i 到 v_j 的路径。

参考答案

（59）C

试题（60）

以下关于哈夫曼树的叙述，正确的是__（60）__。

（60）A. 哈夫曼树一定是满二叉树，其每层结点数都达到最大值

B. 哈夫曼树一定是平衡二叉树，其每个结点左右子树的高度差为-1、0 或 1

C. 哈夫曼树中左孩子结点的权值小于父结点、右孩子结点的权值大于父结点

D. 哈夫曼树中叶子结点的权值越小则距离树根越远、叶子结点的权值越大则距离树根越近

试题（60）分析

本题考查数据结构基础知识。

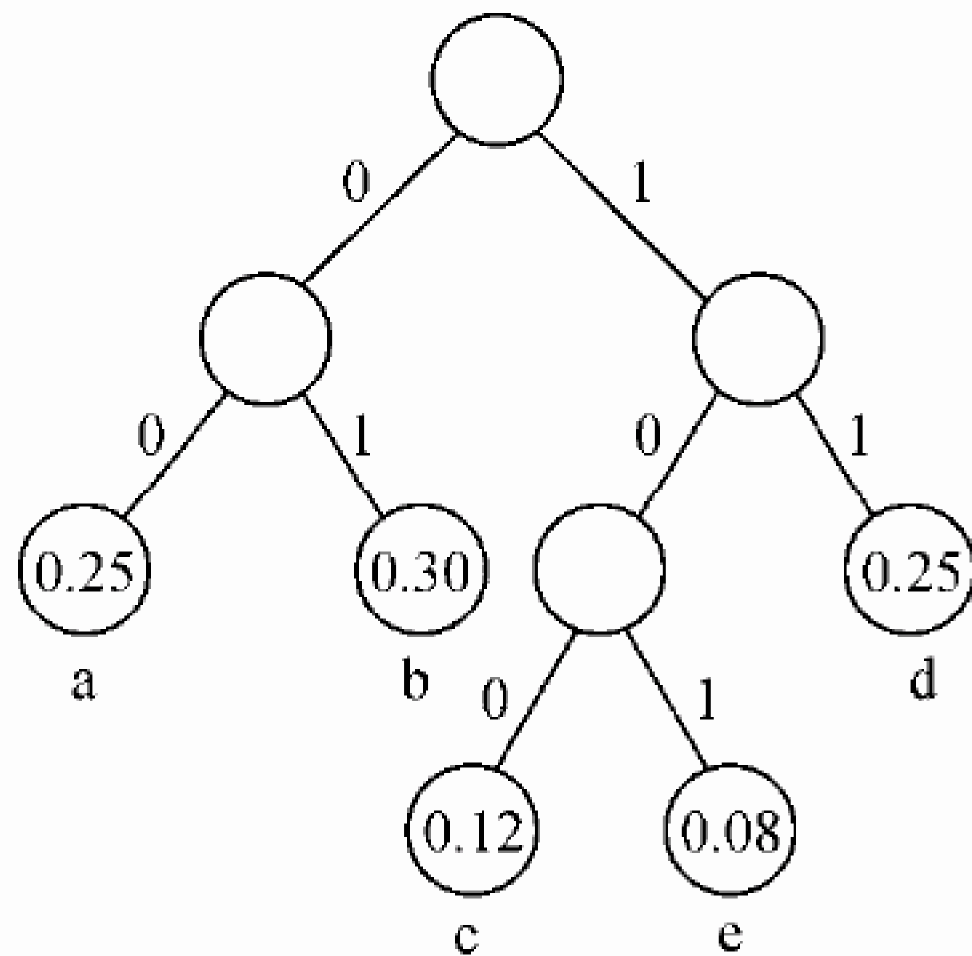
哈夫曼树是一类带权路径长度最短的树，根据一组权值构造出来。构造过程为：

① 根据给定的 n 个权值{w1，w2，…，wn}，构成 n 棵二叉树的集合 F={T1，T2，…，Tn}，其中每棵树 Ti 中只有一个带权为 wi 的根结点，其左右子树均空。

② 在 F 中选取两棵权值最小的树作为左、右子树构造一棵新的二叉树，置新构造二叉树的根结点的权值为其左、右子树根结点的权值之和。

③ 从 F 中删除这两棵树，同时将新得到的二叉树加入到 F 中。

根据权值集合{0.25,0.30,0.08,0.25,0.12}构造的哈夫曼树如下图所示，从中可以知道，哈夫曼树中叶子结点的权值越小则距离树根越远、叶子结点的权值越大则距离树根越近。



参考答案

（60）D

试题（61）

某哈希表（散列表）的长度为 n，设散列函数为 H(Key)=Key mod p，采用线性探测法解决冲突。以下关于 p 值的叙述中，正确的是 （61） 。

（61）A. p 的值一般为不大于 n 且最接近 n 的质数

B. p 的值一般为大于 n 的任意整数

C. p 的值必须为小于 n 的合数

D. p 的值必须等于 n

试题（61）分析

本题考查数据结构基础知识。

在应用散列函数构造哈希表（或散列表）时，由于设计散列函数的目标是：作为一个压缩映像函数，它应具有较大的压缩性，以节省存储空间；哈希函数应具有较好的散

列性，虽然冲突是不可避免的，但应尽量减少。题中所给是常用的除留余数法，p 值一般为不大于 n 且最接近 n 的质数。

参考答案

（61）A

试题（62）、（63）

对 n 个基本有序的整数进行排序，若采用插入排序算法，则时间和空间复杂度分别为＿（62）＿；若采用快速排序算法，则时间和空间复杂度分别为＿（63）＿。

（62）A. $O(n^2)$和 O(n)　　B. O(n)和 O(n)

C. $O(n^2)$和 O(1)　　D. O(n)和 O(1)

（63）A. $O(n^2)$和 O(n)　　B. O(nlgn)和 O(n)

C. $O(n^2)$和 O(1)　　D. O(nlgn)和 O(1)

试题（62～63）分析

本题考查算法分析的基础知识。排序和查找是基本的计算问题，存在很多相关的算法，不同的算法适用于不同的场合。不同的数据输入特点相同的算法也有不同的计算时间。若数据基本有序，对插入排序算法而言，则可以在近似线性时间内完成排序，即 O(n)；而对于快速排序而已，则是其最坏情况，需要二次时间才能完成排序，即 $O(n^2)$。两个算法在排序时仅需要一个额外的存储空间，即空间复杂度均为常数时间复杂度 O(1)。

参考答案

（62）D　（63）C

试题（64）、（65）

在求解某问题时，经过分析发现该问题具有最优子结构性质，求解过程中子问题被重复求解，则采用＿（64）＿算法设计策略；若定义问题的解空间，以深度优先的方式搜索解空间，则采用＿（65）＿算法设计策略。

（64）A. 分治　　B. 动态规划　　C. 贪心　　D. 回溯

（65）A. 动态规划　　B. 贪心　　C. 回溯　　D. 分支限界

试题（64）、（65）分析

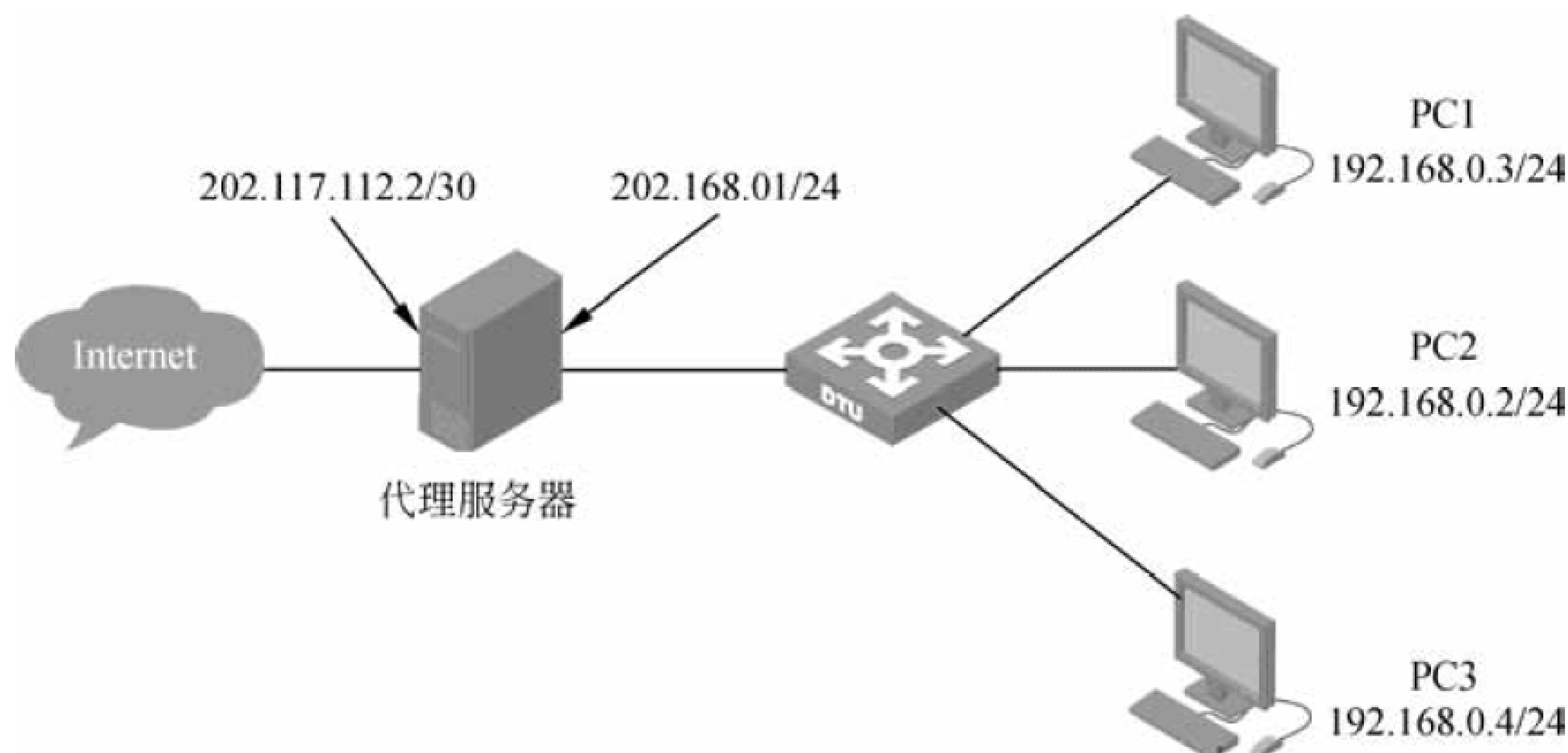
本题考查算法设计的基础知识。存在几种常用的算法设计策略：分治法、动态规划、贪心、回溯发和分支限界法等。其中，分治法一般用于将大问题分解为一个或多个规模较小的子问题，通常采用自顶向下的递归方法来求解。动态规划求解问题的特征是，问题具有最优子结构和重叠子问题，求解时一般采用自底向上的方法来进行。贪心法求解问题的特征是，问题具有最有子结构和贪心选择性质，求解时可以用自底向上或自顶向下的方法进行。回溯法和分支限界法是系统搜索解空间来求解问题的方法，一般先定义解空间，前者以深度优先的方式搜索，后者通常以广度优先的方式搜索。

参考答案

（64）B　　（65）C

试题（66）

某单位的局域网配置如下图所示， PC2 发送到 Internet 上的报文的源 IP 地址为（66）。



（66）A．192.168.0.2　　B．192.168.0.1
　　C．202.117.112.1　　D．202.117.112.2

试题（66）分析

本试题考查局域网配置中 IP 地址设置相关问题。

PC2 发送到 Internet 上的报文经代理服务器转换后，源 IP 地址变成代理服务器的出口 IP 地址，即 202.117.112.2。

参考答案

（66）D

试题（67）、（68）

在 IPv4 向 IPv6 过渡期间，如果要使得两个 IPv6 结点可以通过现有的 IPv4 网络进行通信，则应该使用（67）；如果要使得纯 IPv6 结点可以与纯 IPv4 结点进行通信，则需要使用（68）。

（67）A. 堆栈技术　　B. 双协议栈技术
　　C. 隧道技术　　D. 翻译技术

（68）A. 堆栈技术　　B. 双协议栈技术
　　C. 隧道技术　　D. 翻译技术

试题（67）、（68）分析

如果要使得两个 IPv6 结点可以通过现有的 IPv4 网络进行通信，则应该使用隧道技术，如果要使得纯 IPv6 结点可以与纯 IPv4 结点进行通信，则需要使用翻译技术。

参考答案

（67）C　（68）D

试题（69）、（70）

POP3 协议采用__（69）__模式进行通信，当客户机需要服务时，客户端软件与 POP3 服务器建立__（70）__连接。

（69）A. Browser/Server　　B. Client/Server
　　C. Peer to Peer　　D. Peer to Server

（70）A. TCP　　B. UDP
　　C. PHP　　D. IP

试题（69）、（70）分析

POP3 协议采用 C/S 模式进行通信，POP3 需要 TCP 连接的支持，当客户机需要服务时，客户端软件与 POP3 服务器建立 TCP 连接。

参考答案

（69）B　（70）A

试题（71）～（75）

There is nothing in this world constant but inconstancy. —SWIFT

Project after project designs a set of algorithms and then plunges into construction of customer-deliverable software on a schedule that demands delivery of the first thing built.

In most projects, the first system built is __（71）__ usable. It may be too slow, too big, awkward to use, or all three. There is no __（72）__ but to start again, smarting but smarter, and build a redesigned version in which these problems are solved. The discard and __（73）__ may be done in one lump, or it may be done piece-by-piece. But all large-system experience shows that it will be done. Where a new system concept or new technology is used, one has to build a system to throw away, for even the best planning is not so omniscient (全知的) as to get it right the first time.

The management question, therefore, is not whether to build a pilot system and throw it away. You will do that. The only question is whether to plan in advance to build a __（74）__, or to promise to deliver the throwaway to customers. Seen this way, the answer is much clearer. Delivering that throwaway to customers buys time, but it does so only at the __（75）__ of agony (极大痛苦) for the user, distraction for the builders while they do the redesign, and a bad reputation for the product that the best redesign will find hard to live down.

Hence plan to throw one away; you will, anyhow.

（71）A. almost　　B. often　　C. usually　　D. barely

（72）A. alternative　　B. need　　C. possibility　　D. solution

（73）A. design　　B. redesign　　C. plan　　D. build

（74）A. throwaway　　B. system　　C. software　　D. product

（75）A. worth　　B. value　　C. cost　　D. invaluable

试题（71）～（75）参考译文

不变只是愿望，变化才是永恒。——SWIFT

一个接一个的软件项目都是一开始设计算法，然后将算法应用到待发布的软件中，接着根据时间进度把第一次开发的产品发布给客户。

对于大多数项目，第一个开发的系统并不适用。它可能太慢、太大、难以使用，或者三者兼有。要解决所有的问题，除了重新开始以外，没有其他的办法——即开发一个更灵巧或者更好的系统。系统的丢弃和重新设计可以一步完成，也可以一块块地实现。所有大型系统的经验都显示，这是必须完成的步骤。而且，新的系统概念或新技术会不断出现，因此开发的系统必须被抛弃，但即使是最优秀的项目计划也不能无所不知地在最开始就解决这些问题。

因此，管理上的问题不再是“是否构建一个实验性的系统，然后抛弃它”，你必须这样做。现在的问题是“是否预先计划抛弃原型的开发，或者是否将该原型发布给用户”。从这个角度看待问题，答案更加清晰。将原型发布给用户，虽然可以获得时间，但是其代价高昂——对于用户，使用极度痛苦；对于重新开发的人员，分散了精力；对于产品，影响了声誉，即使是最好的再设计也难以挽回名声。

因此，为舍弃而计划，无论如何，你一定要这样做。

参考答案

（71）D　（72）A　（73）B　（74）A　（75）C

第 20 章　2013 下半年软件设计师下午试题分析与解答

试题一（共 15 分）

阅读下列说明和图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

【说明】

某大学欲开发一个基于 Web 的课程注册系统，该系统的主要功能如下：

1．验证输入信息

（1）检查学生信息：检查学生输入的所有注册所需信息。如果信息不合法，返回学生信息不合法提示；如果合法，输出合法学生信息。

（2）检查学位考试结果：检查学生提供的学位考试结果。如果不合法，返回学位考试结果不合法提示；如果合法，检查该学生注册资格。

（3）检查学生注册资格：根据合法学生信息和合法学位考试结果，检查该学生对欲选课程的注册资格。如果无资格，返回无注册资格提示；如果有注册资格，则输出注册学生信息（包含选课学生标识）和欲注册课程信息。

2．处理注册申请

（1）存储注册信息：将注册学生信息记录在学生库。

（2）存储所注册课程：将选课学生标识与欲注册课程进行关联，然后存入课程库。

（3）发送注册通知：从学生库中读取注册学生信息，从课程库中读取所注册课程信息，给学生发送接受提示；给教务人员发送所注册课程信息和已注册学生信息。

现采用结构化方法对课程注册系统进行分析与设计，获得如图 1-1 所示的 0 层数据流图和图 1-2 所示的 1 层数据流图。

【问题 1】（2 分）

使用说明中的词语，给出图 1-1 中的实体 E1 和 E2 的名称。

【问题 2】（2 分）

使用说明中的词语，给出图 1-2 中的数据存储 D1 和 D2 的名称。

【问题 3】（8 分）

根据说明和图中术语，补充图 1-2 中缺失的数据流及其起点和终点。

【问题 4】（3 分）

根据补充完整的图 1-1 和图 1-2，说明上层的哪些数据流是由下层的哪些数据流组合而成。

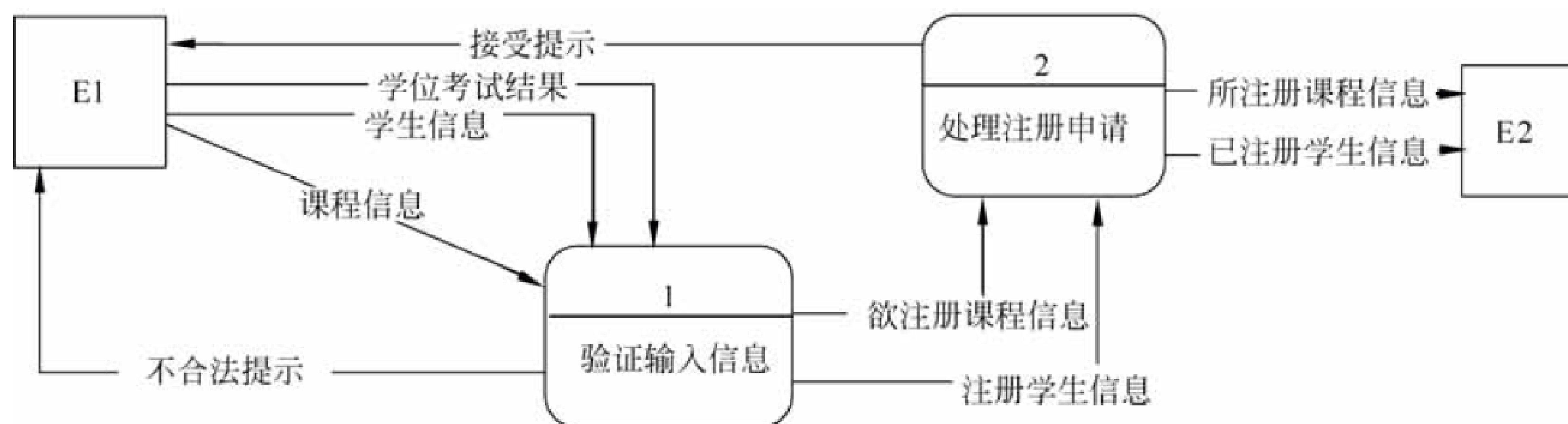


图 1-1 0 层数据流图

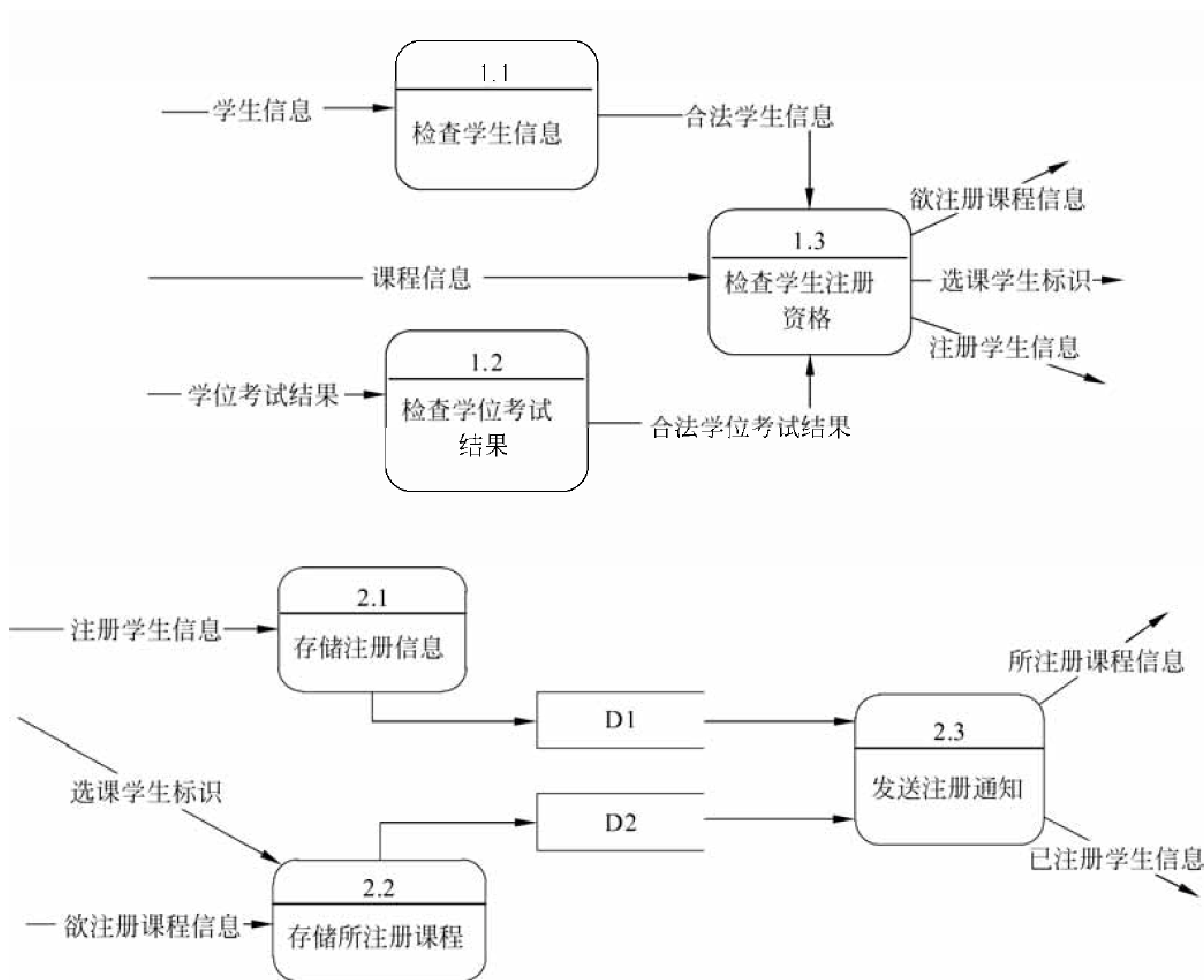


图 1-2 1 层数据流图

试题一分析

本题考查 DFD 的应用，属于比较传统的题目，考查点也与往年类似。

本题目案例说明相对简单，没有从顶层开始给出 DFD，而是直接从 0 层 DFD 开始。

顶层 DFD 通常用来确定系统边界，其中只包含一个唯一的加工（即待开发的系统）、外部实体以及外部实体与系统之间的输入和输出数据流。

【问题 1】

本问题考查 0 层 DFD，要求确定外部实体。不难看出，在 0 层 DFD 中，系统主要功能“验证输入信息”和“处理注册申请”，涉及与系统交互的外部实体有“学生”提供输入信息，发送注册通知功能给“教务人员”发送所注册的课程信息和已注册的学生信息，从而即可确定 E1 为“学生”实体，E2 为“教务人员”实体。

【问题 2】

本问题要求确定 1 层数据流图中的数据存储。分析说明中和数据存储有关的描述，不难发现，说明 2.（1）存储注册信息明确说明“将注册学生信息记录在学生库”，可知 D1 为学生库；说明 2.（2）存储所注册课程中明确说明“然后存入课程库”，可知 D2 为课程库。

【问题 3】

本问题要求补充缺失的数据流及其起点和终点。细心的考生可能会发现，对照图 1-1 和图 1-2 的输入数据流，数量和名称均相同，所以缺失的数据流是输出数据流或者处理之间的数据流。考查图 1-1 中输出至 E1 的数据流，有“接受提示”和“不合法提示”，而图 1-2 中没有这两条数据流，可以确定缺失的数据流包括这两条或者其分解的数据流。

考查说明 1 中的 3 个子功能，（1）检查学生信息完成检查学生输入的所有注册所需信息。如果信息不合法，返回学生信息不合法提示。（2）检查学位考试结果完成检查学生提供的学位考试结果。如果不合法，返回学位考试结果不合法提示。（3）检查学生注册资格完成根据合法学生信息和合法学位考试结果，检查该学生对欲选课程的注册资格。如果无资格，返回无注册资格提示。对应图 1-1 中的处理 1 验证输入信息的输出数据流“不合法提示”，不难发现，在图 1-2 中，处理 1.1 缺少了到实体学生的输出数据流“学生信息不合法提示”；处理 1.2 缺少了到实体学生的输出数据流“无注册资格提示”；处理 1.3 缺少了到实体学生的输出数据流“学位考试结果不合法提示”。

再考查图 1-1 中处理 2，其输出数据流有三条，而图 1-2 中对图 1-1 中处理 2 的分解中，只包含了“所注册课程信息”和“已注册学生信息”两条数据流，缺失了“接受提示”。说明 2.（3）中发送注册通知功能完成从学生库中读取注册学生信息，从课程库中读取所注册课程信息，给学生发送接受提示；给教务人员发送所注册课程信息和已注册学生信息。所以，缺失的“接受提示”的起点是处理 2.3 发送注册通知，终点是 E1 学生。

【问题 4】

本问题考查数据流的分解与组合。仔细分析【说明】中的文字并与图 1-1 的对照，可以发现在图 1-1 中不合法提示在图 1-2 中没有出现。事实上，从前述【问题 3】缺失数据流的分析中，已经发现，图 1-2 中对于说明中的功能出现了“学生信息不合法提示”、“无注册资格提示”和“学位考试结果不合法提示”三条数据流，说明图 1-1 中的数据流

“不合法提示”是由这三条数据流组合而成。同样，说明 2.（2）存储所注册课程并将选课学生标识与欲注册课程进行关联，然后存入课程库，图 1-1 中注册学生信息在图 1-2 中进一步分出注册学生信息和选课学生标识，即图 1-1 中注册学生信息是注册学生信息和选课学生标识的并集。

参考答案

【问题 1】

E1：学生　　　　E2：教务人员

【问题 2】

D1：学生库　　　　D2：课程库

【问题 3】

数　据　流	起　　点	终　　点
学生信息不合法提示	1.1 或 检查学生信息	E1 或 学生
无注册资格提示	1.3 或 检查学生注册资格	E1 或 学生
学位考试结果不合法提示	1.2 或 检查学位考试结果	E1 或 学生
接受提示	2.3 或 发送注册通知	E1 或 学生

【问题 4】

图 1-1 中不合法提示分解为图 1-2 中的三条数据流的组合：学生信息不合法提示、无注册资格提示、学位考试结果不合法提示。

图 1-1 中注册学生信息对应图 1-2 中注册学生信息和选课学生标识。

试题二（共 15 分）

阅读下列说明，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某快递公司为了方便管理公司物品运送的各项业务活动，需要构建一个物品运送信息管理系统。

【需求分析结果】

（1）快递公司有多个分公司，分公司信息包括分公司编号、名称、经理、办公电话和地址。每个分公司可以有多名员工处理分公司的日常业务，每名员工只能在一个分公司工作。每个分公司由一名经理负责管理分公司的业务和员工，系统需要记录每个经理的任职时间。

（2）员工信息包括员工号、姓名、岗位、薪资、手机号和家庭地址。其中，员工号唯一标识员工信息的每一个元组。岗位包括经理、调度员、业务员等。业务员根据客户提交的快件申请单进行快件受理事宜，一个业务员可以受理多个客户的快件申请，一个快件申请只能由一个业务员受理。调度员根据已受理的申请单安排快件的承运事宜，例如：执行承运的业务员、运达时间等。一个业务员可以执行调度员安排的多个快件的承

运业务。

（3）客户信息包括客户号、单位名称、通信地址、所属省份、联系人、联系电话、银行账号。其中，客户号唯一标识客户信息的每一个元组。当客户要寄快件时，先要提交快件申请单，申请号由系统自动生成。快件申请信息包括申请号、客户号、发件人、发件人电话、快件名称、运费、发出地、收件人、收件人电话、收件地址。其中，一个申请号对应唯一的一个快件申请，一个客户可以提交多个快件申请，但一个快件申请由唯一的一个客户提交。

【概念模型设计】

根据需求阶段收集的信息，设计的实体联系图（图 2-1）和关系模式（不完整）如下：

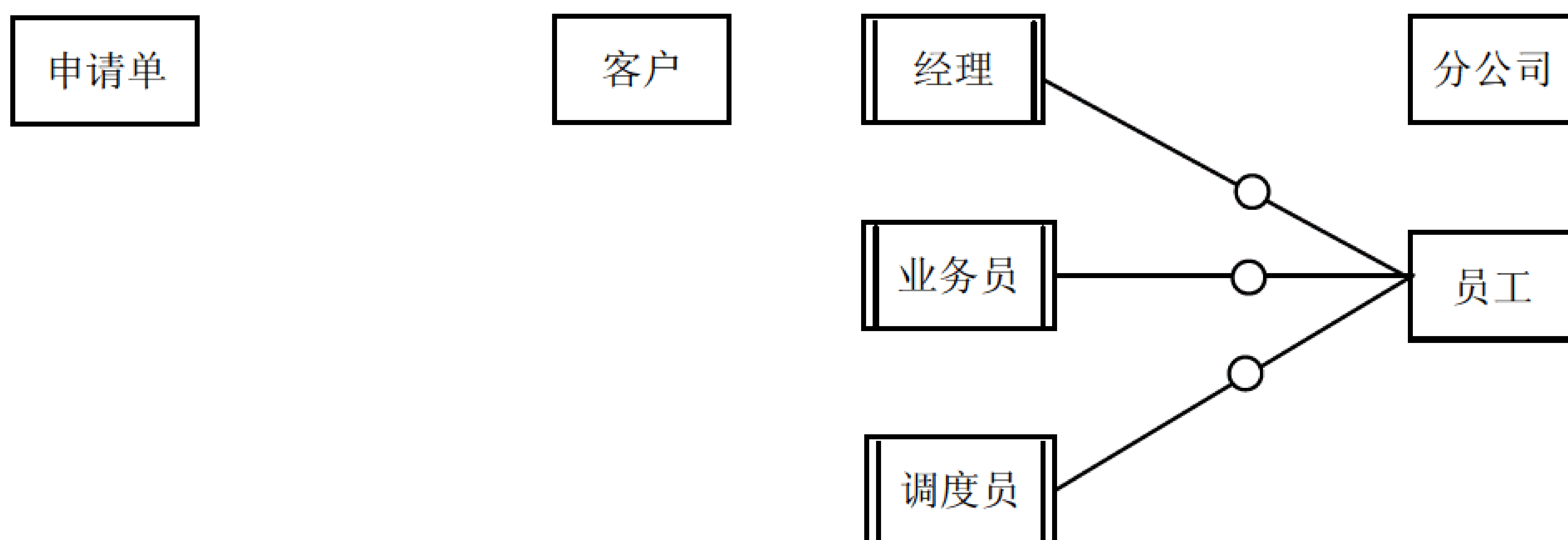


图 2-1　实体联系图

【关系模式设计】

分公司（分公司编号，名称，经理，办公电话，地址）

员工（员工号，姓名，____(a)____，岗位，薪资，手机号，家庭地址）

客户（客户号，单位名称，通信地址，所属省份，联系人，联系电话，银行账号）

申请单（____(b)____，发件人，发件人电话，发件人地址，快件名称，运费，收件人，收件人电话，收件地址，受理标志，业务员）

安排承运（____(c)____，实际完成时间，调度员）

【问题 1】（5 分）

根据问题描述，补充五个联系，完善图 2-1 的实体联系图。联系名可用联系 1、联系 2、联系 3、联系 4 和联系 5 代替，联系的类型分为 1:1、1:n 和 m:n（或 1:1、1:*和*:*）。

【问题 2】（6 分）

（1）根据实体联系图，将关系模式中的空（a）～（c）补充完整。

（2）给出员工、申请单和安排承运关系模式的主键和外键。

【问题 3】（4 分）

（1）客户关系的通信地址可以进一步分为邮编、省、市、街道，那么该属性是否属

于简单属性，为什么？请用 100 字以内的文字说明。

（2）假设分公司需要增设一位经理的职位，那么分公司与经理之间的联系类型应修改为___（d）___，分公司的主键应修改为___（e）___。

试题二分析

本题考查数据库概念结构设计及概念结构向逻辑结构转换的过程。

此类题目要求考生认真阅读题目对现实问题的描述，经过分类、聚集、概括等方法，从中确定实体及其联系。题目已经给出了 4 个实体，需要根据需求描述，给出实体间的联系。

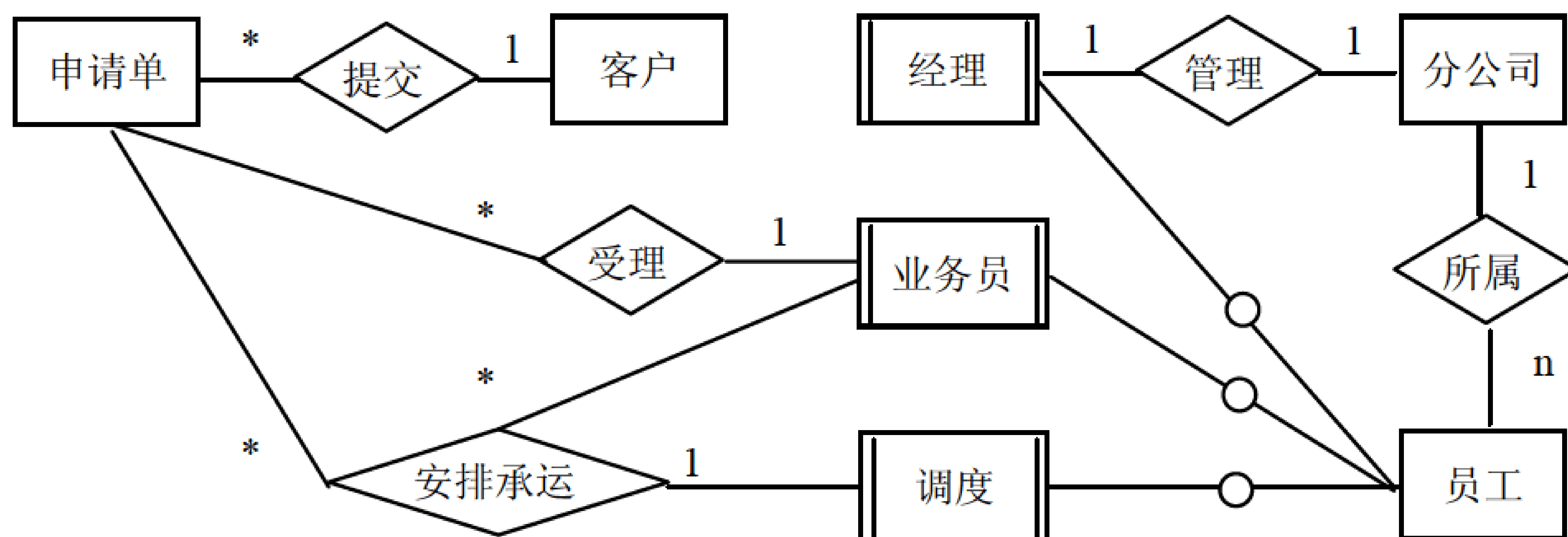
【问题 1】

由“每个分公司有一位经理”可知分公司与经理之间的管理联系类型为 1:1；由“每个分公司有多名员工处理日常事务，每个员工属于一个分公司”可知分公司与员工间的所属联系类型为 1:*；并且员工是经理的超类型，经理是员工的子类型。

由“一个客户可以有多个快件申请，但一个快件申请对应唯一的一个客户”可知，客户与申请单之间的提交联系类型为 1:*。

由“业务员根据客户提交的快件申请单进行快件受理事宜，一个业务员可以受理多个客户的快件申请，一个快件申请只能由一个业务员受理”可知业务员与申请单之间的受理联系类型为 1:*。

由“调度根据已受理的申请单安排快件的承运事宜，例如：执行承运的业务员、运达时间等；一个业务员可以执行调度安排的多个快件的承运业务。”可知，调度、业务员和申请单之间的承运联系类型为 1:*:*。



【问题 2】

（1）完整的关系模式如下：

分公司（分公司编号，名称，经理，办公电话，地址）

员工（员工号，姓名，分公司编号，岗位，薪资，手机号，家庭地址）

客户（客户号，单位名称，通信地址，所属省份，联系人，联系电话，银行账号）

申请单（申请号，客户号，发件人，发件人电话，发件人地址，快件名称，运费，

收件人，收件人电话，收件地址，受理标志，业务员）

安排承运（申请号，业务员，实际完成时间，调度员）

（2）员工、申请单和安排承运关系模式的主键和外键的分析如下：

在申请单信息中，申请号由系统自动生成，不会重复，可作为申请单的主键属性，外键为客户号，业务员；在员工信息中，员工号唯一标识员工信息的每一个元组，故为员工关系的主键属性，外键为分公司编号；安排承运关系模式的主键为申请号，外键为业务员和调度员。

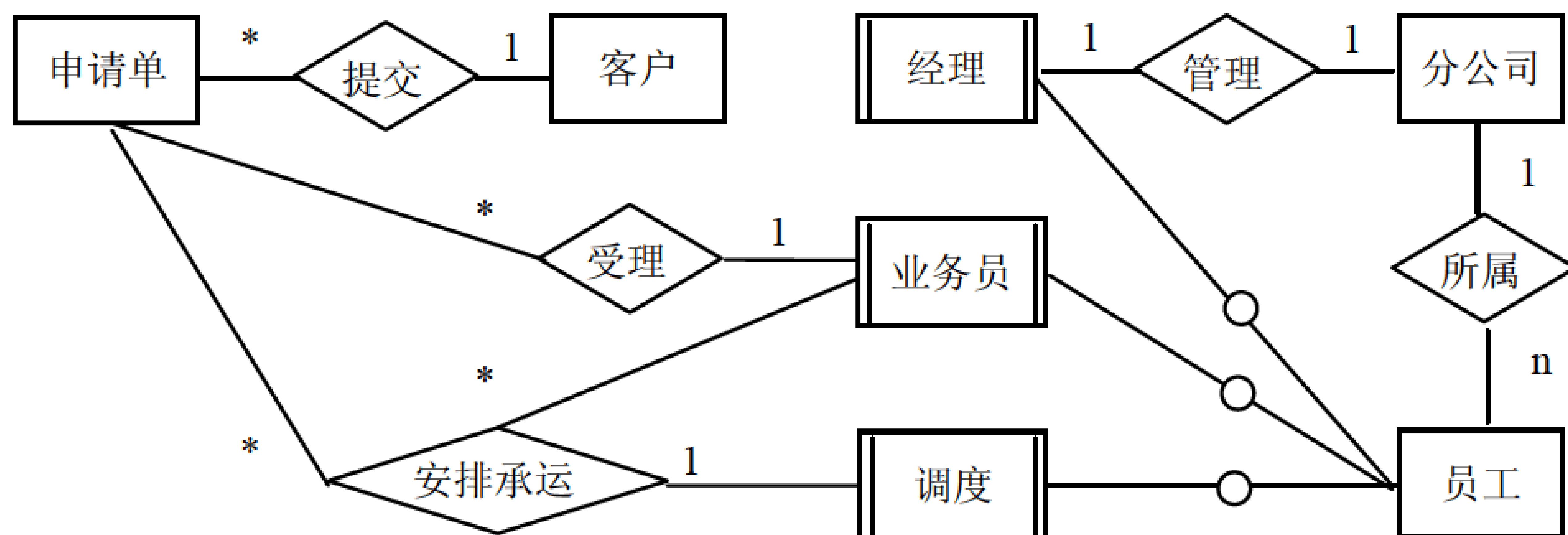
【问题 3】

（1）客户的通信地址属性不属于简单属性。因为根据题意，客户关系的通信地址可以进一步分为邮编、省、市、街道，而简单属性是原子的、不可再分的，复合属性可以细分为更小的部分（即划分为别的属性）。由于客户的通信地址可以进一步分为邮编、省、市、街道，故属于复合属性。

（2）根据题意，分公司需要增设一位经理的职位，那么分公司与经理之间的联系类型应修改为 1:n，分公司的主键应修改为分公司编号，经理。

参考答案

【问题 1】图中的*可表示为 m 或 n，对联系名称可不做要求，但不能出现重名。



【问题 2】

（1）（a）分公司编号

（b）申请号，客户号

（c）申请号，业务员

（2）

关 系 模 式	主　　键	外　　键
员工	员工号	分公司编号
申请单	申请号	客户号，业务员
安排承运	申请号	业务员，调度员

【问题 3】

（1）该属性不属于简单属性。因为简单属性是原子的、不可再分的，复合属性是可

以细分为更小的部分（即划分为别的属性），本题客户关系的通信地址可以进一步分为邮编、省、市、街道，所以属于复合属性。

（2）（d）1:n

（e）分公司编号，经理

试题三（共 15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某航空公司会员积分系统（CFrequentFlyer）的主要功能描述如下：

乘客只要办理该航空公司的会员卡，即可成为普卡会员（CBasic）。随着飞行里程数的积累，可以从普卡会员升级到银卡会员（CSilver）或金卡会员（CGold）。非会员（CNonMember）不能累积里程数。

每年年末，系统根据会员在本年度累积的里程数对下一年会员等级进行调整。

普卡会员在一年内累积的里程数若满 25,000 英里但不足 50,000 英里，则自动升级为银卡会员；若累积的里程数在 50,000 英里以上，则自动升级为金卡会员。银卡会员在一年内累积的里程数若在 50,000 英里以上，则自动升级为金卡会员。

若一年内没有达到对应级别要求的里程数，则自动降低会员等级。金卡会员一年内累积的里程数若不足 25,000 英里，则自动降级为普卡会员；若累积的里程数达到 25,000 英里，但是不足 50,000 英里，则自动降级为银卡会员。银卡会员一年内累积的里程数若不足 25,000 英里，则自动降级为普卡会员。

采用面向对象方法对会员积分系统进行分析与设计，得到如图 3-1 所示的状态图和图 3-2 所示的类图。

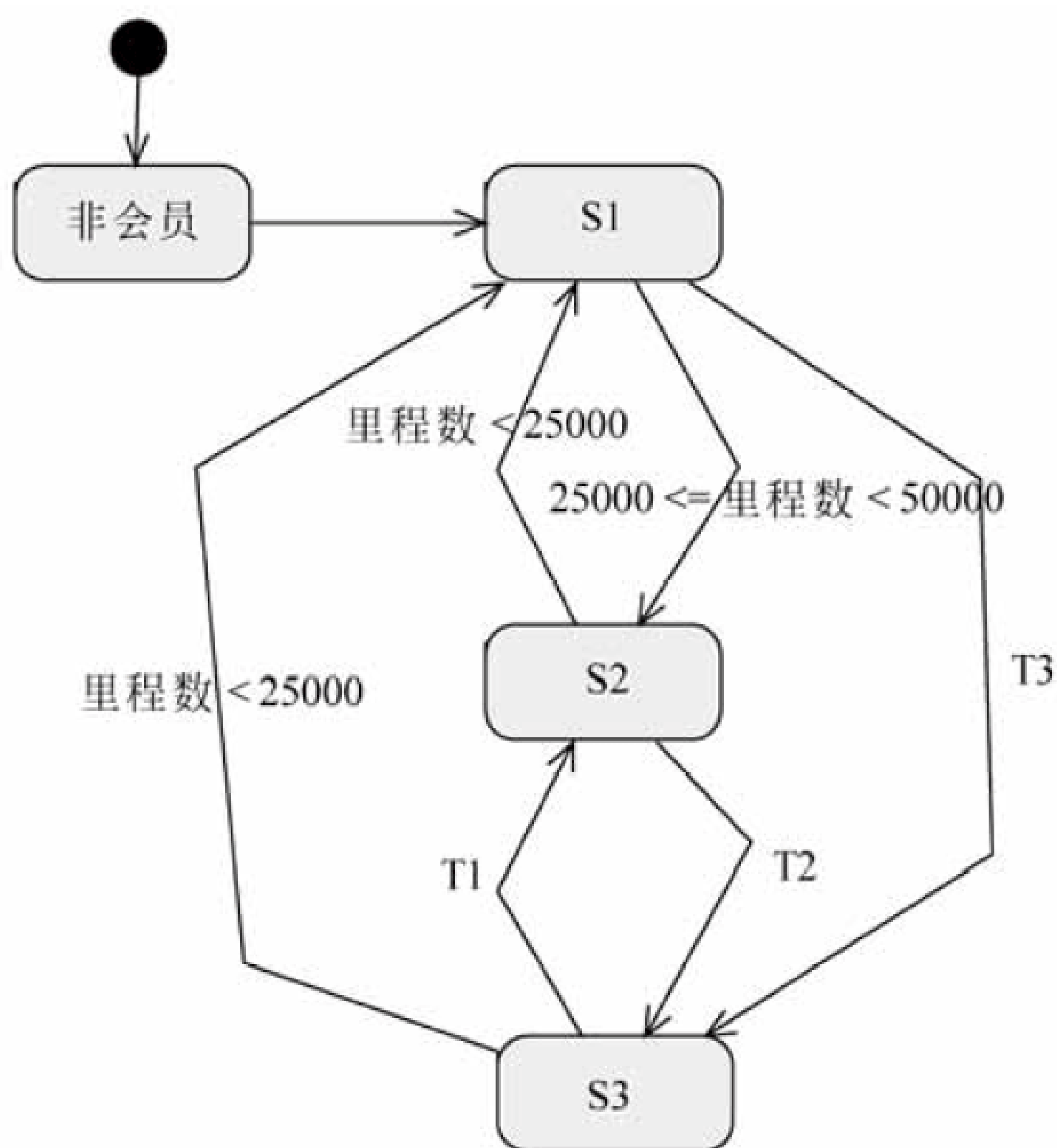


图 3-1 状态图

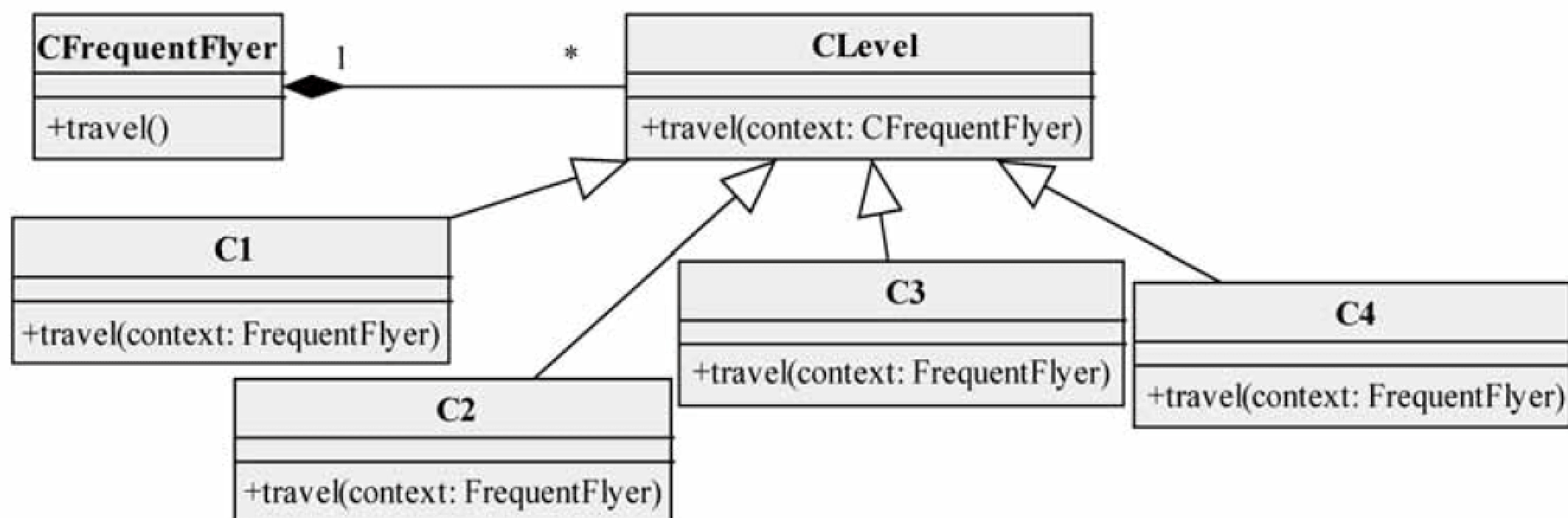


图 3-2　类图

【问题 1】（6 分）

根据说明中的描述，给出图 3-1 中 S1~S3 处所对应的状态以及 T1~T3 处所对应的迁移的名称。

【问题 2】（4 分）

根据说明中的描述，给出图 3-2 中 C1～C4 所对应的类名（类名使用说明中给出的英文词汇）。

【问题 3】（5 分）

图 3-2 所示的类图中使用了哪种设计模式？在这种设计模式下，类 CFrequentFlyer 必须具有的属性是什么？C1～C4 中的 travel 方法应具有什么功能？

试题三分析

本题属于经典的考题，主要考查面向对象分析方法以及 UML 的状态图和类图的相关知识及应用。

【问题 1】

UML 中的状态图主要用于描述一个对象在其生存期间的动态行为，表现一个对象所经历的装填序列，引起状态转移的事件以及因状态转移而伴随的动作。图中给出的是会员的状态图。图中要求填充 S1、S2、S3 这三个状态以及它们之间的变迁关系。本题中会员有三种状态：普卡、金卡和银卡。根据说明，办理会员卡之后即可成为普卡会员，所以 S1 可以判定为普卡会员。当“里程数满 25,000 英里但不足 50,000 英里，则自动升级为银卡会员”，所以 S2 应为银卡会员，那么 S3 就应该是金卡会员。T1、T2 就是 S2 和 S3 之间的转换原则。T3 是 S1->S2 的转换原则。由说明可知，S2->S3（T2）：里程数在 50,000 英里以上；S3->S3（T1）：里程数达到 25,000 英里，但是不足 50,000 英里；S1->S3（T3）：累积的里程数在 50,000 英里以上。

【问题 2】

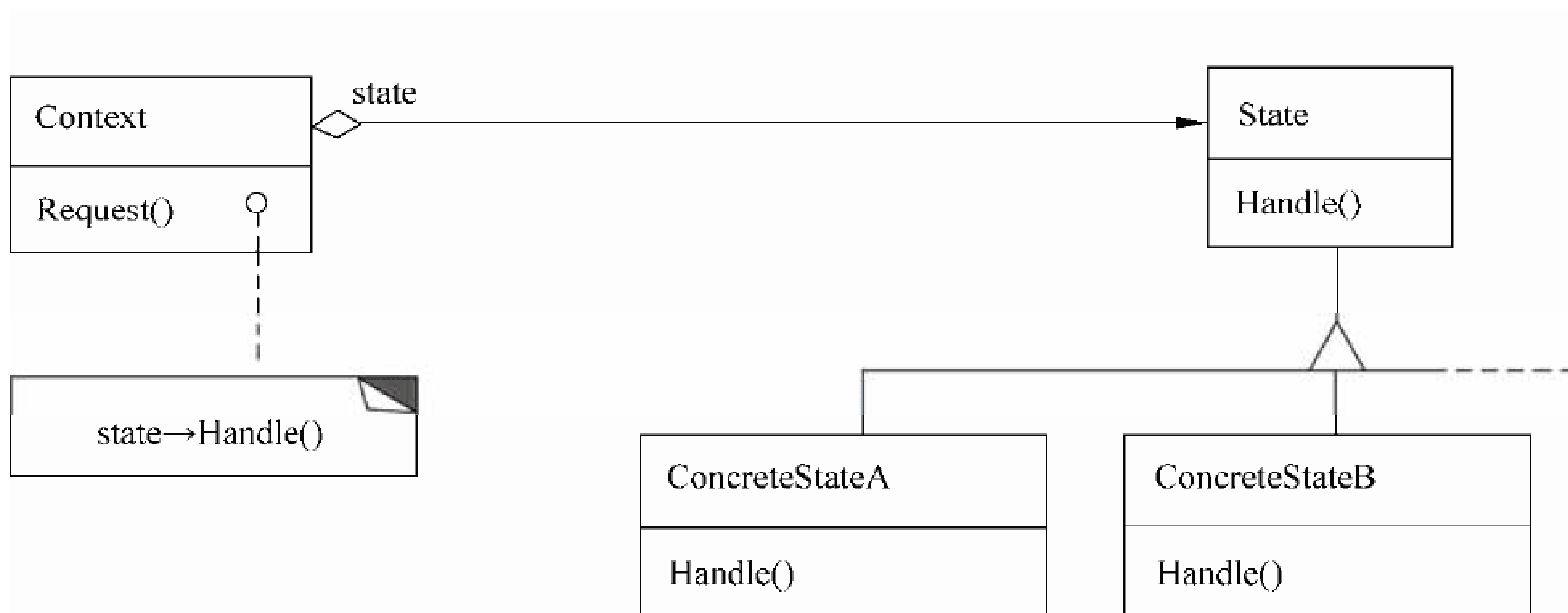
由图 3-2 可知，需要补充的是继承结构中的子类。根据题目说明，能够具有一般/

特殊关系的只有不同级别的会员。所以 C1~C4 依次应该是：CNonMember、CBasic、CSilver、CGold。

【问题 3】

本题在设计类时使用到了状态模式。

状态模式允许对象在内部状态变化时，变更其行为，并且修改其类。状态模式的类图如下所示。



其中：

- 环境类（Context）：定义客户感兴趣的接口。维护一个 ConcreteState 子类的实例，这个实例定义当前状态。
- 抽象状态类（State）：定义一个接口以封装与 Context 的一个特定状态相关的行为。
- 具体状态类（ConcreteState）：每一子类实现一个与 Context 的一个状态相关的行为。

图 3-2 中的类 CFrequentFlyer 对应上图中的环境类，因此类 CFrequentFlyer 应该有一个 CLevel 类的对象。

travel 方法的功能：计算飞行里程数，根据里程数判断是否需要调整会员级别（跳转到不同的状态）。

参考答案

【问题 1】

S1：普卡、普卡会员

S2：银卡、银卡会员

S3：金卡、金卡会员

T1：25000 <= 里程数 <50000　　T2：里程数 >= 50000　　T3：里程数 >= 50000

【问题 2】

C1：CNonMember C2：CBasic　　C3：CSilver　　C4：CGold

（C1~C4 的次序可以互换）

【问题 3】

使用了 State 模式（状态模式）。

类 CFrequentFlyer 必须具有的属性：CLevel 的对象。

travel 方法的功能：计算飞行里程数，根据里程数判断是否需要调整会员级别（跳转到不同的状态）。

试题四（共 15 分）

阅读下列说明和 C 代码，回答问题 1 至问题 3，将解答写在答题纸的对应栏内。

【说明】

某工程计算中要完成多个矩阵相乘（链乘）的计算任务。

两个矩阵相乘要求第一个矩阵的列数等于第二个矩阵的行数，计算量主要由进行乘法运算的次数决定。采用标准的矩阵相乘算法，计算 $A_{m\times n}*B_{n\times p}$，需要 m*n*p 次乘法运算。

矩阵相乘满足结合律，多个矩阵相乘，不同的计算顺序会产生不同的计算量。以矩阵 $A1_{10\times100}$，$A2_{100\times5}$，$A3_{5\times50}$ 三个矩阵相乘为例，若按（A1*A2）*A3 计算，则需要进行 10*100*5+10*5*50=7500 次乘法运算；若按 A1*（A2*A3）计算，则需要进行 100*5*50+10*100*50 = 75000 次乘法运算。可见不同的计算顺序对计算量有很大的影响。

矩阵链乘问题可描述为：给定 n 个矩阵<A1,A2,…,An>，矩阵 Ai 的维数为 $p_{i-1}\times p_i$，其中 i=1,2,…,n。确定一种乘法顺序，使得这 n 个矩阵相乘时进行乘法的运算次数最少。

由于可能的计算顺序数量非常庞大，对较大的 n，用蛮力法确定计算顺序是不实际的。经过对问题进行分析，发现矩阵链乘问题具有最优子结构，即若 A1*A2*…*An 的一个最优计算顺序从第 k 个矩阵处断开，即分为 A1*A2*…*Ak 和 Ak+1*Ak+2*…*An 两个子问题，则该最优解应该包含 A1*A2*…*Ak 的一个最优计算顺序和 Ak+1*Ak+2*…*An 的一个最优计算顺序。据此构造递归式，

$$\cos t[i][j]=\begin{cases}0 & \text{if } i=j\\ \min_{i\le k<j}\cos t[i][k]+\cos t[k+1][j]+p_i*p_{k+1}*p_{j+1} & \text{if } i<j\end{cases}$$

其中，cost[i][j]表示 Ai+1*Ai+2*…*Aj+1 的最优计算的计算代价。最终需要求解 cost[0][n-1]。

【C 代码】

算法实现采用自底向上的计算过程。首先计算两个矩阵相乘的计算量，然后依次计算 3 个矩阵、4 个矩阵……n 个矩阵相乘的最小计算量及最优计算顺序。下面是该算法的 C 语言实现。

（1）主要变量说明

n：矩阵数

seq[]：矩阵维数序列

cost[][]：二维数组，长度为n*n，其中元素cost[i][j]表示Ai+1*Ai+2*…*Aj+1的最优计算的计算代价

trace[][]：二维数组，长度为n*n，其中元素trace[i][j]表示Ai+1*Ai+2*…*Aj+1的最优计算对应的划分位置，即k

（2）函数cmm

```
#define    N 100
int cost[N][N];
int trace[N][N];
int cmm(int n, int seq[]){
    int tempCost;
    int tempTrace;
    int i, j, k, p;
    int temp;
    for(i = 0; i < n; i++){  cost[i][i] = 0;    }
    for(p = 1; p < n; p++){
        for(i = 0; ___(1)___; i++){
            ___(2)___;
            tempCost = -1;
            for(k = i; k < j; k++){
            temp = ___(3)___;
                if(tempCost == -1 || tempCost > temp){
                    tempCost = temp;
                    ___(4)___;
                }
            }
            cost[i][j] = tempCost;
            trace[i][j] = tempTrace;
        }
    }
    return cost[0][n - 1];
}
```

【问题1】（8分）

根据以上说明和C代码，填充C代码中的空（1）～（4）。

【问题2】（4分）

根据以上说明和C代码，该问题采用了___（5）___算法设计策略，时间复杂度为___（6）___（用O符号表示）。

【问题 3】（3 分）

考虑实例 n = 6，各个矩阵的维数：A1 为 5*10，A2 为 10*3，A3 为 3*12，A4 为 12*5，A5 为 5*50，A6 为 50*6，即维数序列为 5, 10, 3, 12, 5, 50, 6。则根据上述 C 代码得到的一个最优计算顺序为___（7）___（用加括号方式表示计算顺序），所需要的乘法运算次数为___（8）___。

试题四分析

本题考查算法设计、分析和 C 程序实现的应用知识，属于传统题目，考查点也与往年类似。

题干部分较清楚的描述了求解矩阵链乘问题的算法，理解递归式是关键，n 个矩阵相乘是一个规模为 n 的问题，从某个 k 处断开，该问题可以分解为两个子问题，即 A1*A2*…*An 分为 A1*A2*…*Ak 和 Ak+1*Ak+2*…*An 两个子问题。求解时采用自底向上的方法。

【问题 1】

本问题考查算法的实现。C 程序中主要部分是三重循环，循环变量 p 定义了求解问题的规模，因为是自底向上，因此，p 的值应该是从 1 到 n-1，即从规模为 1 的问题一直到规模为 n-1 的问题。循环变量 i 是要求解的子问题的起始，从 0 开始，最大为 n-p-1，故（1）处应填 n-p。确定了 i 和 p 之后，下来就要确定 j 了，显然，空（2）处为 j=i+p。循环变量 k 是问题 Ai*Ai+1*…*Aj 的划分位置，对每一个 k，都要计算需要的计算成本，可以根据递归式来填写，空（3）处为 cost[i][k] + cost[k + 1][j] + seq[i] * seq[k + 1] * seq[j +1]。确定每个问题 Ai*Ai+1*…*Aj 的划分位置 k 之后，要把这个 k 值记住，放在变量 tempTrace 中，即空（4）处填写 tempTrace = k。

【问题 2】

本问题考查算法的设计策略和时间复杂度，从题干说明可以很容易看出，问题具有最优子结构和重叠子问题，采用自底向上的方法求解，这些都是动态规划的典型特点，因此采用的是动态规划设计策略。从上述 C 程序很容易分析出，程序中没有递归，存在三重循环，故时间复杂度为 $O(n^3)$。

【问题 3】

本问题考查算法的应用。通过一个具体实例可以更容易理解问题和求解方法。可以根据问题 1 中的程序执行来求解。启发式的思路是先把维度最大的消掉，如 A5*A6 相乘之后，维度 50 就没有了，所以考虑这两个矩阵先相乘；然后是 A3*A4 相乘之后，维度 12 就没有了，所以考虑这两个矩阵相乘；接着，A1*A2 相乘之后，维度 10 就没有了，所以考虑这两个矩阵相乘……这样可以确定相乘的顺序((A1A2) ((A3A4)(A5A6)))，需要的计算开销分别是 5* 50* 6 = 1500, 3 * 12 * 5 = 180, 5 * 10 * 3 = 150， 3 * 5 * 6 = 90， 5 * 3 * 6 = 90，把上述值相加，即 1500 + 180 + 150 + 90 + 90 = 2010。

参考答案

【问题 1】

（1）i < n – p

（2）j = i + p

（3）cost[i][k] + cost[k + 1][j] + seq[i] * seq[k + 1] * seq[j + 1]

（4）tempTrace = k

【问题 2】

（5）动态规划

（6）$O(n^3)$

【问题 3】

（7）((A1A2) ((A3A4)(A5A6)))

（8）2010

试题五（共 15 分）（试题五和试题六任选一道）

阅读下列说明和 C++代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

欲开发一个绘图软件，要求使用不同的绘图程序绘制不同的图形。以绘制直线和圆形为例，对应的绘图程序如表 5-1 所示。

表 5-1 不同的绘图程序

	DP1	DP2
绘制直线	draw_a_line(x1,y1,x2,y2)	drawline(x1,x2,y1,y2)
绘制圆	draw_a_circle(x, y, r)	drawcircle(x, y, r)

该绘图软件的扩展性要求，将不断扩充新的图形和新的绘图程序。为了避免出现类爆炸的情况，现采用桥接（Bridge）模式来实现上述要求，得到如图 5-1 所示的类图。

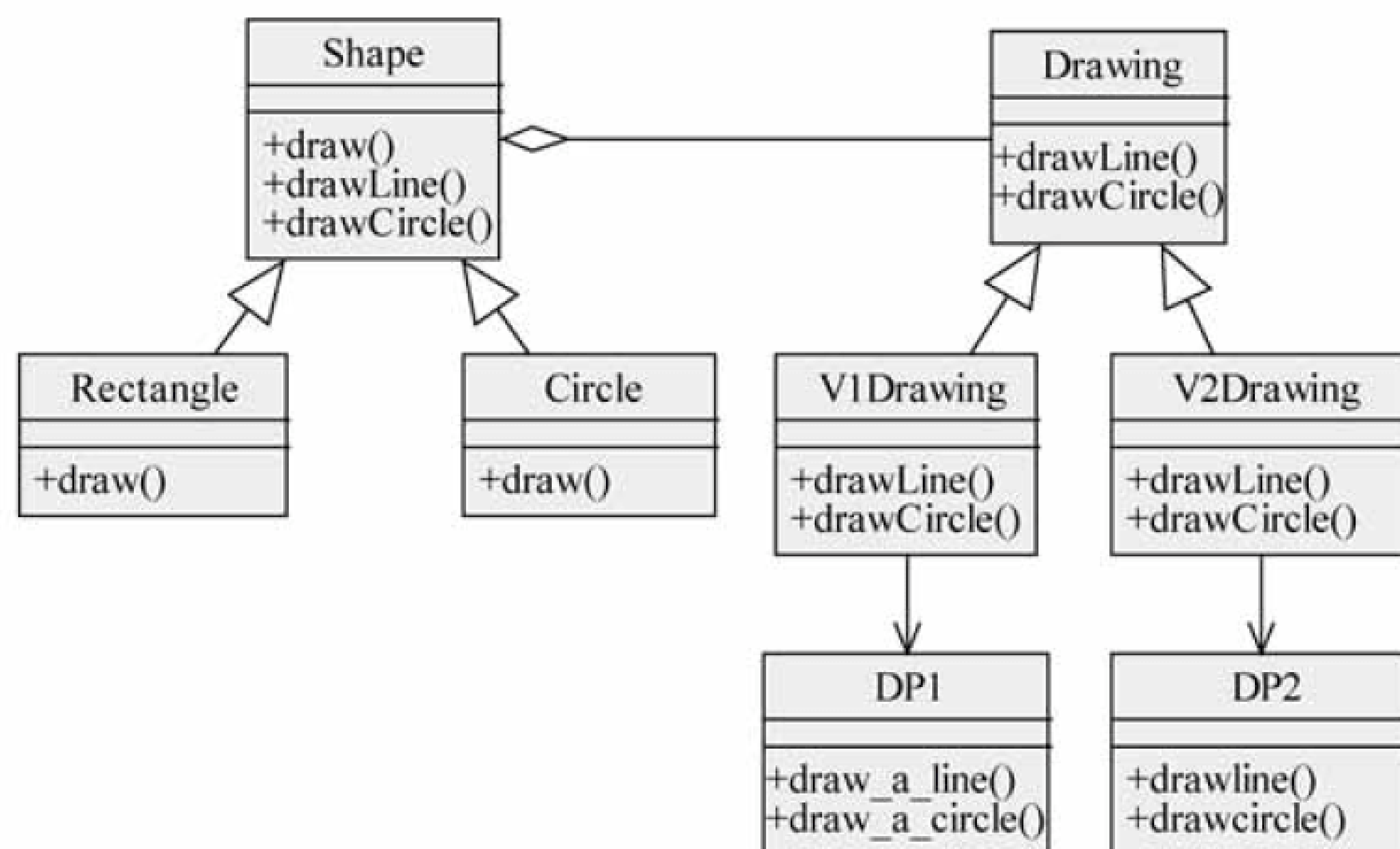


图 5-1 类图

【C++代码】

```
class DP1 {
public:
   static void draw_a_line(double x1, double y1, double x2, double y2) { /*
代码省略 */   }
   static void draw_a_circle(double x, double y, double r) {  /* 代码省略
*/   }
};
class DP2 {
public:
   static void drawline(double x1, double x2, double y1, double y2) {  /*
代码省略 */   }
   static void drawcircle(double x, double y, double r) {  /* 代码省略 */   }
};
class Drawing {
public:
   ____(1)____;
   ____(2)____;
};
class V1Drawing : public Drawing {
public:
    void drawLine(double x1, double y1, double x2, double y2) { /* 代码
省略 */ }
    void drawCircle(double x, double y, double r) {  ____(3)____;   }
};
class V2Drawing : public Drawing {
public:
    void drawLine(double x1, double y1, double x2, double y2)  { /* 代码
省略 */ }
    void drawCircle(double x, double y, double r) {  ____(4)____;   }
};
class Shape {
public:
   ____(5)____;
   Shape(Drawing *dp) { _dp = dp; }
   void drawLine(double x1, double y1, double x2, double y2) {
        _dp->drawLine(x1, y1, x2, y2);   }
   void drawCircle(double x, double y, double r) { _dp->drawCircle(x, y,
r); }
private:  Drawing *_dp;
};
```

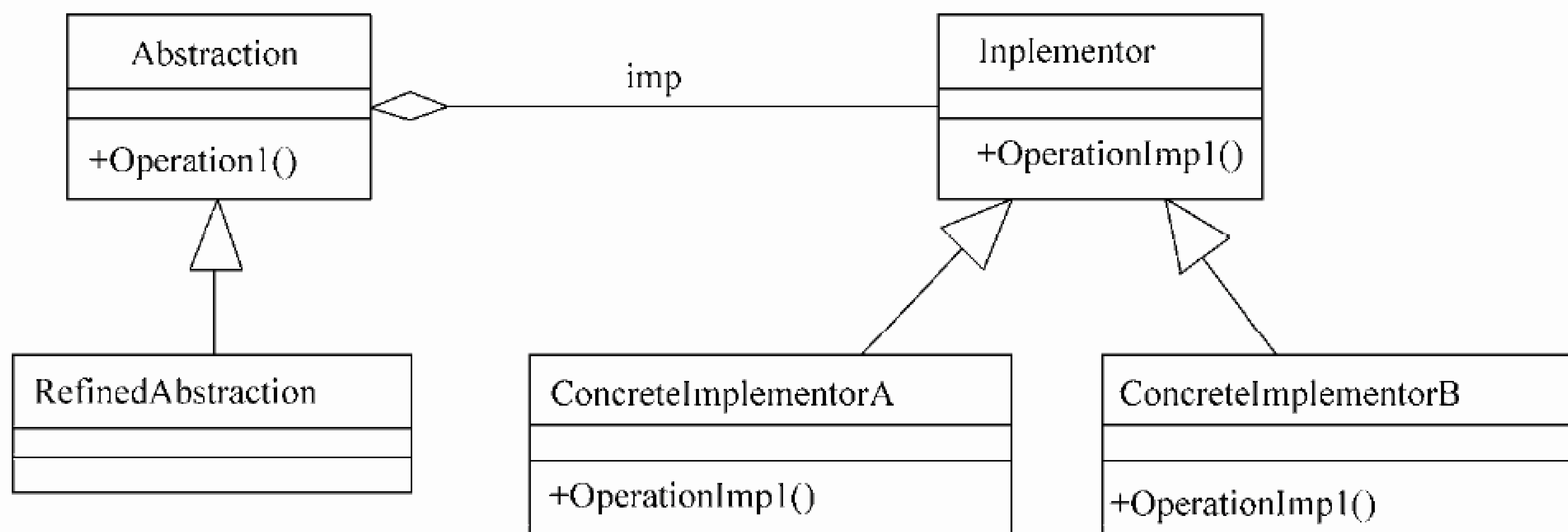
```
class Rectangle : public Shape {
public:
  void draw() {  /* 代码省略 */   }
// 其余代码省略
};

class Circle : public Shape {
private:  double _x, _y, _r;
public:
  Circle(Drawing *dp, double x, double y, double r) :___(6)___ { _x =
x; _y = y; _r = r; }
  void draw() { drawCircle(_x, _y, _r); }
};
```

试题五分析

本题考查桥接（Bridge）模式的概念及应用。

Bridge 模式可以将复杂的组件分成两个独立的、但又相关的继承层次结构：功能性的抽象和内部实现。改变组件的这两个层次结构很简单，以至于它们可以相互独立地变化。当具有抽象的层次结构和相应的实现层次结构时，Bridge 模式是非常有用的。除了可以将抽象和实现组合成许多不同的类，该模式还可以动态组合的独立类的形式实现这些抽象和实现。下图所示是 Bridge 模式的类图。



在以下情况中，应该使用 Bridge 模式：

- 想避免在抽象及其实现之间存在永久的绑定；
- 抽象及其实现可以使用子类进行扩展；
- 抽象的实现被改动应该对客户端没有影响，也就是说，不需要重新编译代码。

本题中，类 Shape 对应上图中的 Abstraction，表示抽象部分；类 Drawing 对应

Implementor，表示实现部分。这两个类的子类分别表示具体的抽象部分和实现部分。在C++中，Drawing 可以用抽象类来实现，将其中的方法定义为纯虚拟函数。因此（1）、（2）分别应为“virtual void drawLine(double x1, double y1, double x2, double y2) = 0”、“virtual void drawCircle(double x, double y, double r) = 0”。

V1Drawing 是绘图实现类之一，它采用的绘图程序由是 DP1 所提供的。DP1 中的方法均为静态方法，必须用类名来引用。因此（3）处应为 DP1::draw_a_circle(x, y, r)。同理（4）处应为“DP2::drawcircle(x, y, r)”。

由类图可以看出，Shape 类中定义的方法 draw 在其子类中被重置了，而 Shape 表示的是抽象部分，可以将 draw 方法定义为纯虚拟函数。所以，（5）应该为“virtual void draw() = 0”。

空（6）处考查继承结构中子类构造函数的定义。构造子类对象时，需要调用基类的构造函数，这可以通过初始化列表显式指明需要调用的基类的构造函数。在本题中，Shape 类只定义了一个构造函数，因此（6）应该为“Shape(dp)”。

参考答案

（1）virtual void drawLine(double x1, double y1, double x2, double y2) = 0

（2）virtual void drawCircle(double x, double y, double r) = 0

（3）DP1::draw_a_circle(x, y, r)

（4）DP2::drawcircle(x, y, r)

（5）virtual void draw() = 0

（6）Shape(dp)

试题六（共 15 分）（试题五和试题六任选一道）

阅读下列说明和 Java 代码，将应填入＿（n）＿处的字句写在答题纸的对应栏内。

【说明】

欲开发一个绘图软件，要求使用不同的绘图程序绘制不同的图形。以绘制直线和圆形为例，对应的绘图程序如表 6-1 所示。

表 6-1　不同的绘图程序

	DP1	DP2
绘制直线	draw_a_line(x1,y1,x2,y2)	drawline(x1,x2,y1,y2)
绘制圆	draw_a_circle(x, y, r)	drawcircle(x, y, r)

该绘图软件的扩展性要求，将不断扩充新的图形和新的绘图程序。为了避免出现类爆炸的情况，现采用桥接（Bridge）模式来实现上述要求，得到如图 6-1 所示的类图。

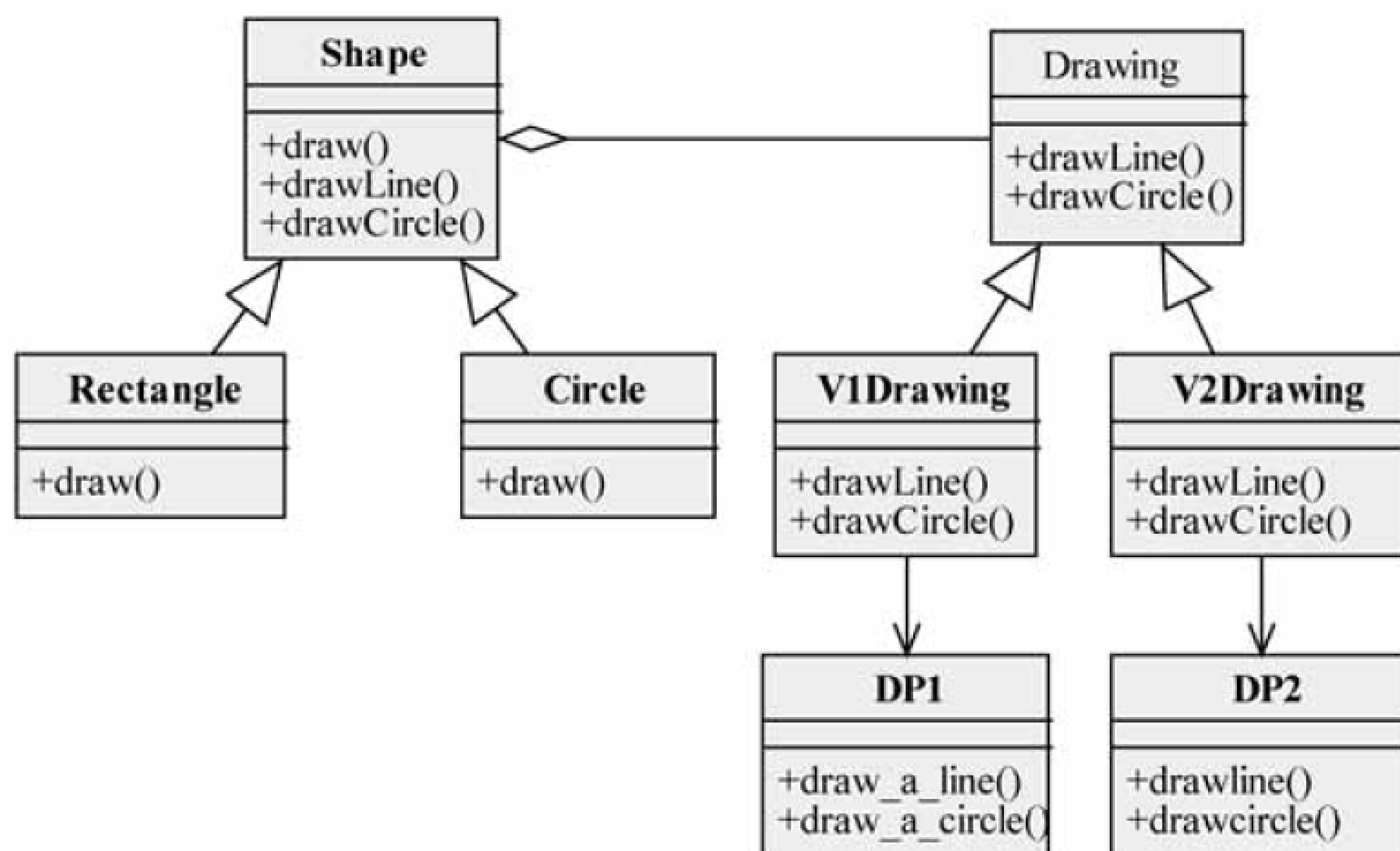


图 6-1　类图

【Java 代码】

```
    (1)     Drawing {
    (2)     ;
    (3)     ;
}

class DP1{
    static public void draw_a_line(double x1, double y1, double x2, double
y2)
{/*代码省略 */ }
    static public void draw_a_circle (double x, double y, double r) { /*
代码省略 */ }
}

class DP2{
    static public void drawline(double x1, double y1, double x2, double y2)
{/*代码省略 */ }
    static public void drawcircle (double x, double y, double r) {/*代码
省略 */ }
}

class V1Drawing implements Drawing {
    public void drawLine(double x1, double y1, double x2, double y2) { /*
代码省略 */ }
```

```
        public void drawCircle(double x, double y, double r) {      (4)      ;  }
    }

    class V2Drawing implements Drawing {
        public void drawLine(double x1, double y1, double x2, double y2) { /*
代码省略 */ }
        public void drawCircle(double x, double y, double r) {      (5)      ;  }
    }

    abstract class Shape {
      private Drawing _dp;
          (6)      ;
      Shape(Drawing dp) { _dp = dp; }
      public void drawLine(double x1, double y1, double x2, double y2) {
          _dp.drawLine(x1, y1, x2, y2);  }
      public void drawCirle(double x, double y, double r) { _dp.drawCircle(x,
y, r);  }
    }

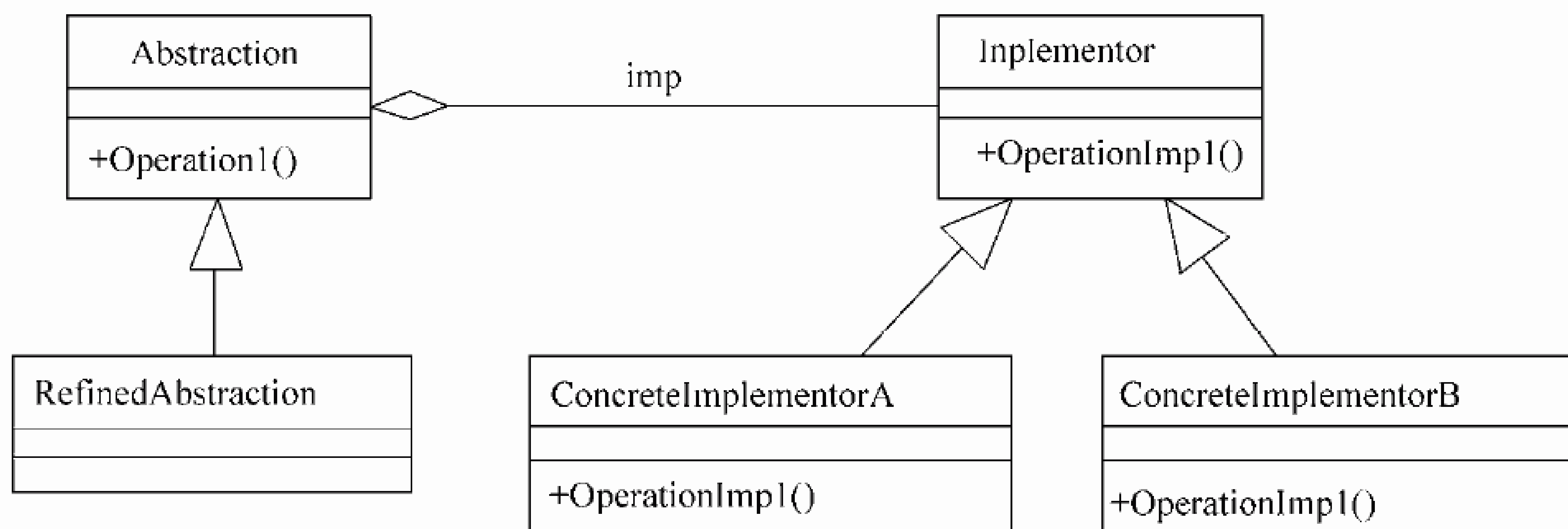
    class Rectangle extends Shape {
        private double _x1, _x2, _y1, _y2;
        public Rectangle (Drawing dp, double x1, double y1, double x2, double
y2)
    { /* 代码省略 */ }
        public void draw() { /* 代码省略 */ }
    }

    class Circle extends Shape {
        private double _x, _y, _r;
        public Circle(Drawing dp, double x, double y, double r)  { /* 代码省略
*/ }
        public void draw() { drawCirle(_x, _y, _r);    }
    }
```

试题六分析

本题考查桥接（Bridge）模式的概念及应用。

Bridge 模式可以将复杂的组件分成两个独立的但又相关的继承层次结构：功能性的抽象和内部实现。改变组件的这两个层次结构很简单，以至于它们可以相互独立地变化。当具有抽象的层次结构和相应的实现层次结构时，Bridge 模式是非常有用的。除了可以将抽象和实现组合成许多不同的类，该模式还可以以动态组合的独立类的形式实现这些抽象和实现。下图所示是 Bridge 模式的类图。



在以下情况中，应该使用 Bridge 模式：

- 想避免在抽象及其实现之间存在永久的绑定；
- 抽象及其实现 可以使用子类进行扩展；
- 抽象的实现被改动应该对客户端没有影响，也就是说，不需要重新编译代码。

本题中，类 Shape 对应上图中的 Abstraction，表示抽象部分；类 Drawing 对应 Implementor，表示实现部分。这两个类的子类分别表示具体的抽象部分和实现部分。类 Drawing 为具体的实现类提供统一接口，在 Java 中可以使用接口来实现。因此（1）、（2）、（3）分别应为“interface”、“void drawLine(double x1, double y1, double x2, double y2)”、“void drawCircle(double x, double y, double r)”。

V1Drawing 是绘图实现类之一，它采用的绘图程序由是 DP1 所提供的。因此（4）处应为“DP1.draw_a_circle(x, y, r)”。同理（5）处应为“DP2.drawcircle(x, y, r)”。

由类图可以看出，Shape 类中定义的方法 draw 在其子类中被重置了，而 Shape 表示的是抽象部分，可以将 draw 方法定义为抽象函数。所以，（6）应该为“abstract public void draw()”。

参考答案

（1）interface

（2）void drawLine(double x1, double y1, double x2, double y2)

（3）void drawCircle(double x, double y, double r)

（4）DP1.draw_a_circle(x, y, r)

（5）DP2.drawcircle(x, y, r)

（6）abstract public void draw()

第 21 章　2014 上半年软件设计师上午试题分析与解答

试题（1）

在 CPU 中，常用来为 ALU 执行算术逻辑运算提供数据并暂存运算结果的寄存器是<u>（1）</u>。

（1）A．程序计数器　　B．状态寄存器
　　C．通用寄存器　　D．累加寄存器

试题（1）分析

本题考查计算机系统基础知识。

CPU 中有一些重要的寄存器，程序计数器（PC）用于存放指令的地址。当程序顺序执行时，每取出一条指令，PC 内容自动增加一个值，指向下一条要取的指令。当程序出现转移时，则将转移地址送入 PC，然后由 PC 给出新的指令地址。

状态寄存器用于记录运算中产生的标志信息。状态寄存器中的每一位单独使用，称为标志位。标志位的取值反映了 ALU 当前的工作状态，可以作为条件转移指令的转移条件。典型的标志位有以下几种：进位标志位（C）、零标志位（Z）、符号标志位（S）、溢出标志位（V）、奇偶标志位（P）。

通用寄存器组是 CPU 中的一组工作寄存器，运算时用于暂存操作数或地址。在程序中使用通用寄存器可以减少访问内存的次数，提高运算速度。累加器（accumulator）是一个数据寄存器，在运算过程中暂时存放操作数和中间运算结果，不能用于长时间地保存一个数据。

参考答案

（1）D

试题（2）

某机器字长为 n，最高位是符号位，其定点整数的最大值为<u>（2）</u>。

（2）A．2^n-1　　B．$2^{n-1}-1$　　C．2^n　　D．2^{n-1}

试题（2）分析

本题考查计算机系统中数据表示基础知识。

机器字长为 n，最高位为符号位，则剩余的 n-1 位用来表示数值，其最大值是这 n-1 位都为 1，也就是 $2^{n-1}-1$。

参考答案

（2）B

试题（3）

海明码利用奇偶性检错和纠错，通过在 n 个数据位之间插入 k 个校验位，扩大数据编码的码距。若 n=48，则 k 应为 （3） 。

（3）A. 4　　B. 5　　C. 6　　D. 7

试题（3）分析

本题考查数据校验基础知识。

设数据位是 n 位，校验位是 k 位，则 n 和 k 必须满足以下关系：$2^k-1 \geqslant n+k$。

若 n=48，则 k 为 6 时可满足 $2^6-1 \geqslant 48+6$。

海明码的编码规则如下。

设 k 个校验位为 P_k，P_{k-1}，…，P_1，n 个数据位为 D_{n-1}，D_{n-2}，…，D_1，D_0，对应的海明码为 H_{n+k}，H_{n+k-1}，…，H_1，那么：

① P_i 在海明码的第 2^{i-1} 位置，即 $H_j=P_i$，且 $j=2^{i-1}$；数据位则依序从低到高占据海明码中剩下的位置。

② 海明码中的任一位都是由若干个校验位来校验的。其对应关系如下：被校验的海明位的下标等于所有参与校验该位的校验位的下标之和，而校验位则由自身校验。

参考答案

（3）C

试题（4）、（5）

通常可以将计算机系统中执行一条指令的过程分为取指令、分析和执行指令 3 步，若取指令时间为 4Δt，分析时间为 2Δt，执行时间为 3Δt，按顺序方式从头到尾执行完 600 条指令所需时间为 （4） Δt；若按照执行第 i 条、分析第 i+1 条、读取第 i+2 条重叠的流水线方式执行指令，则从头到尾执行完 600 条指令所需时间为 （5） Δt。

（4）A. 2400　　B. 3000　　C. 3600　　D. 5400

（5）A. 2400　　B. 2405　　C. 3000　　D. 3009

试题（4）、（5）分析

本题考查指令系统基础知识。

指令顺序执行时，每条指令需要 9Δt（4Δt＋2Δt＋3Δt），执行完 600 条指令需要 5400Δt，若采用流水方式，则在分析和执行第 1 条指令时，就可以读取第 2 条指令，当第 1 条指令执行完成，第 2 条指令进行分析和执行，而第 3 条指令可进行读取操作。因此，第 1 条指令执行完成后，每 4Δt 就可以完成 1 条指令，600 条指令的总执行时间为 9Δt＋599×4Δt=2405Δt。

参考答案

（4）D　　（5）B

试题（6）

若用 256K×8bit 的存储器芯片，构成地址 40000000H 到 400FFFFFH 且按字节编址

的内存区域，则需＿（6）＿片芯片。

（6）A．4　　B．8　　C．16　　D．32

试题（6）分析

本题考查计算机系统中存储器知识。

地址 400000000H 到 4000FFFFFH 共 FFFFFH（即 2^{20}）个以字节为单位的编址单元，而 256K×8bit 的存储器芯片可提供 2^{18} 个以字节为单位的编址单元，因此需要 4 片（$2^{20}/2^{18}$）这种芯片来构成上述内存区域。

参考答案

（6）A

试题（7）

以下关于木马程序的叙述中，正确的是＿（7）＿。

（7）A．木马程序主要通过移动磁盘传播

B．木马程序的客户端运行在攻击者的机器上

C．木马程序的目的是使计算机或网络无法提供正常的服务

D．Sniffer 是典型的木马程序

试题（7）分析

本题考查木马程序的基础知识。

木马程序一般分为服务器端（Server）和客户端（Client），服务器端是攻击者传到目标机器上的部分，用来在目标机上监听等待客户端连接过来。客户端是用来控制目标机器的部分，放在攻击者的机器上。

木马（Trojans）程序常被伪装成工具程序或游戏，一旦用户打开了带有特洛伊木马程序的邮件附件或从网上直接下载，或执行了这些程序之后，当你连接到互联网上时，这个程序就会通知黑客用户的 IP 地址及被预先设定的端口。黑客在收到这些资料后，再利用这个潜伏其中的程序，就可以恣意修改用户的计算机设定、复制任何文件、窥视用户整个硬盘内的资料等，从而达到控制用户的计算机的目的。

现在有许多这样的程序，国外的此类软件有 Back Office、Netbus 等，国内的此类软件有 Netspy、YAI、SubSeven、冰河、“广外女生”等。Sniffer 是一种基于被动侦听原理的网络分析软件。使用这种软件，可以监视网络的状态、数据流动情况以及网络上传输的信息，其不属于木马程序。

参考答案

（7）B

试题（8）

防火墙的工作层次是决定防火墙效率及安全的主要因素，以下叙述中，正确的是＿（8）＿。

（8）A．防火墙工作层次越低，工作效率越高，安全性越高

B．防火墙工作层次越低，工作效率越低，安全性越低

C．防火墙工作层次越高，工作效率越高，安全性越低

D．防火墙工作层次越高，工作效率越低，安全性越高

试题（8）分析

本题考查防火墙的基础知识。

防火墙的性能及特点主要由以下两方面所决定。

① 工作层次。这是决定防火墙效率及安全的主要因素。一般来说，工作层次越低，则工作效率越高，但安全性就低了；反之，工作层次越高，工作效率越低，则安全性越高。

② 防火墙采用的机制。如果采用代理机制，则防火墙具有内部信息隐藏的特点，相对而言，安全性高，效率低；如果采用过滤机制，则效率高，安全性却降低了。

参考答案

（8）D

试题（9）

以下关于包过滤防火墙和代理服务防火墙的叙述中，正确的是<u>（9）</u>。

（9）A．包过滤技术实现成本较高，所以安全性能高

B．包过滤技术对应用和用户是透明的

C．代理服务技术安全性较高，可以提高网络整体性能

D．代理服务技术只能配置成用户认证后才建立连接

试题（9）分析

本题考查防火墙的基础知识。

显然，包过滤防火墙采用包过滤技术对应用和用户是透明的。

参考答案

（9）B

试题（10）

王某买了一幅美术作品原件，则他享有该美术作品的<u>（10）</u>。

（10）A．著作权　　B．所有权

C．展览权　　D．所有权与其展览权

试题（10）分析

本题考查知识产权基本知识。

绘画、书法、雕塑等美术作品的原件可以买卖、赠与。但获得一件美术作品并不意味着获得该作品的著作权。我国著作权法规定："美术等作品原件所有权的转移，不视为作品著作权的转移，但美术作品原件的展览权由原件所有人享有。"这就是说作品物转移的事实并不引起作品著作权的转移，受让人只是取得物的所有权和作品原件的展览权，作品的著作权仍然由作者享有。

参考答案

（10）D

试题（11）

甲、乙两软件公司于 2012 年 7 月 12 日就其财务软件产品分别申请“用友”和“用有”商标注册。两财务软件相似，甲第一次使用时间为 2009 年 7 月，乙第一次使用时间为 2009 年 5 月。此情形下，__（11）__能获准注册。

（11）A. “用友”　　B. “用友”与“用有”都
　　C. “用有”　　D. 由甲、乙抽签结果确定谁

试题（11）分析

我国商标注册采取“申请在先”的审查原则，当两个或两个以上申请人在同一种或者类似商品上申请注册相同或者近似商标时，商标主管机关根据申请时间的先后，决定商标权的归属，申请在先的人可以获得注册。对于同日申请的情况，使用在先的人可以获得注册。如果同日使用或均未使用，则采取申请人之间协商解决，协商不成的，由各申请人抽签决定。

参考答案

（11）C

试题（12）、（13）

以下媒体中，__（12）__是表示媒体，__（13）__是表现媒体。

（12）A. 图像　　B. 图像编码　　C. 电磁波　　D. 鼠标

（13）A. 图像　　B. 图像编码　　C. 电磁波　　D. 鼠标

试题（12）、（13）分析

本题考查多媒体基础知识。

国际电话电报咨询委员会（CCITT）将媒体分为感觉媒体、表示媒体、表现媒体、存储媒体和传输媒体 5 类，其中感觉媒体指直接作用于人的感觉器官，使人产生直接感觉的媒体，如引起听觉反应的声音，引起视觉反应的图像等；传输媒体指传输表示媒体的物理介质，如电缆、光缆、电磁波等；表示媒体指传输感觉媒体的中介媒体，即用于数据交换的编码，如图像编码、文本编码和声音编码等；表现媒体是指进行信息输入和输出的媒体，如键盘、鼠标、话筒，以及显示器、打印机、喇叭等；存储媒体指用于存储表示媒体的物理介质，如硬盘、光盘等。

参考答案

（12）B　（13）D

试题（14）

__（14）__表示显示器在横向（行）上具有的像素点数目。

（14）A. 显示分辨率　　B. 水平分辨率
　　C. 垂直分辨率　　D. 显示深度

试题（14）分析

本题考查多媒体基础知识。

显示分辨率是指显示器上能够显示出的像素点数目，即显示器在横向和纵向上能够显示出的像素点数目。水平分辨率表明显示器水平方向（横向）上显示出的像素点数目，垂直分辨率表明显示器垂直方向（纵向）上显示出的像素点数目。例如，显示分辨率为1024×768 则表明显示器水平方向上显示 1024 个像素点，垂直方向上显示 768 个像素点，整个显示屏就含有 796432 个像素点。屏幕能够显示的像素越多，说明显示设备的分辨率越高，显示的图像质量越高。显示深度是指显示器上显示每个像素点颜色的二进制位数。

参考答案

（14）B

试题（15）

以下关于结构化开发方法的叙述中，不正确的是＿（15）＿。

（15）A．将数据流映射为软件系统的模块结构

B．一般情况下，数据流类型包括变换流型和事务流型

C．不同类型的数据流有不同的映射方法

D．一个软件系统只有一种数据流类型

试题（15）分析

本题考查结构化开发方法的结构化设计。

结构化设计方法是一种面向数据流的设计方法，与结构化分析方法衔接。在需求分析阶段，结构化分析方法产生了数据流图，而在设计阶段，结构化设计方法将数据流映射为软件系统的模块结构。数据流图中从系统的输入数据流到系统的输出数据流的一连串变换形成了一条信息流。其中的信息流一般情况下包括变换流型和事物流型。不同类型的数据流到程序模块的映射方法不同。一个软件系统往往不仅仅有一种数据流类型。

参考答案

（15）D

试题（16）

模块 A 提供某个班级某门课程的成绩给模块 B，模块 B 计算平均成绩、最高分和最低分，将计算结果返回给模块 A，则模块 B 在软件结构图中属于＿（16）＿模块。

（16）A．传入　B．传出　C．变换　D．协调

试题（16）分析

本题考查结构化开发方法的基础知识。

通常，可以按照在软件系统中的功能将模块分为四种类型。①传入模块：取得数据或输入数据，经过某些处理，再将其传送给其他模块。②传出模块：输出数据，在输出之前可能进行某些处理，数据可能被输出到系统的外部，或者会输出到其他模块进行进一步处理。③变换模块：从上级调用模块得到数据，进行特定的处理，转换成其他形式，

在将加工结果返回给调用模块。④协调模块 一般不对数据进行加工，主要是通过调用、协调和管理其他模块来完成特定的功能。

参考答案

（16）C

试题（17）

（17） 软件成本估算模型是一种静态单变量模型，用于对整个软件系统进行估算。

（17）A．Putnam　　　　B．基本 COCOMO
　　　C．中级 COCOMO　　D．详细 COCOMO

试题（17）分析

本题考查软件项目管理的基础知识。

Putnam 和 COCOMO 都是软件成本估算模型。Putnam 模型是一种动态多变量模型，假设在软件开发的整个生存期中工作量有特定的分布。结构性成本模型 COCOMO 模型分为基本 COCOMO 模型、中级 COCOMO 模型和详细 COCOMO。基本 COCOMO 模型是一个静态单变量模型，对整个软件系统进行估算；中级 COCOMO 模型是一个静态多变量模型，将软件系统模型分为系统和部件两个层次，系统由部件构成；详细 COCOMO 模型将软件系统模型分为系统、子系统和模块三个层次，除了包括中级模型所考虑的因素外，还考虑了在需求分析、软件设计等每一步的成本驱动属性的影响。

参考答案

（17）B

试题（18）

以下关于进度管理工具 Gantt 图的叙述中，不正确的是 （18） 。

（18）A．能清晰地表达每个任务的开始时间、结束时间和持续时间
　　　B．能清晰地表达任务之间的并行关系
　　　C．不能清晰地确定任务之间的依赖关系
　　　D．能清晰地确定影响进度的关键任务

试题（18）分析

本题考查软件项目管理的基础知识。

Gantt 图是一种简单的水平条形图，以日历为基准描述项目任务。水平轴表示日历时间线，如日、周和月等，每个条形表示一个任务，任务名称垂直的列在左边的列中，图中水平条的起点和终点对应水平轴上的时间，分别表示该任务的开始时间和结束时间，水平条的长度表示完成该任务所持续的时间。当日历中同一时段存在多个水平条时，表示任务之间的并发。

Gantt 图能清晰地描述每个任务从何时开始，到何时结束，任务的进展情况以及各个任务之间的并行性。但它不能清晰地反映出各任务之间的依赖关系，难以确定整个项目的关键所在，也不能反映计划中有潜力的部分。

参考答案

（18）D

试题（19）

项目复杂性、规模和结构的不确定性属于＿（19）＿风险。

（19）A．项目　　B．技术　　C．经济　　D．商业

试题（19）分析

本题考查软件项目管理的基础知识。

项目经理需要尽早预测项目中的风险，这样就可以制订有效的风险管理计划以减少风险的影响，所以，早期的风险识别是非常重要的。一般来说，影响软件项目的风险主要有三种类别：项目风险涉及各种形式的预算、进度、人员、资源以及和客户相关的问题；技术风险涉及到潜在的设计、实现、对接、测试即维护问题；业务风险包括建立一个无人想要的优秀产品的风险、失去预算或人员承诺的风险等；商业风险包括如市场风险、策略风险、管理风险和预算风险等。

参考答案

（19）A

试题（20）

以下程序设计语言中，＿（20）＿更适合用来进行动态网页处理。

（20）A．HTML　　B．LISP　　C．PHP　　D．Java/C++

试题（20）分析

本题考查程序语言基础知识。

网页文件本身是一种文本文件，通过在其中添加标记符，可以告诉浏览器如何显示其中的内容。HTML 是超文本标记语言，超文本是指页面内可以包含图片、链接，甚至音乐、程序等非文字元素。

PHP（超文本预处理器）是一种通用开源脚本语言，它将程序嵌入到 HTML 文档中去执行，从而产生动态网页。

参考答案

（20）C

试题（21）

在引用调用方式下进行函数调用，是将＿（21）＿。

（21）A．实参的值传递给形参　　B．实参的地址传递给形参
　　C．形参的值传递给实参　　D．形参的地址传递给实参

试题（21）分析

本题考查程序语言基础知识。

值调用和引用调用是实现函数调用时传递参数的两种基本方式。在值调用方式下，是将实参的值传给形参，在引用调用方式下，是将实参的地址传递给形参。

参考答案

（21）B

试题（22）

编译程序对高级语言源程序进行编译的过程中，要不断收集、记录和使用源程序中一些相关符号的类型和特征等信息，并将其存入__（22）__中。

（22）A. 符号表　B. 哈希表　C. 动态查找表　D. 栈和队列

试题（22）分析

本题考查程序语言基础知识。

编译是实现高级程序设计语言的一种方式，编译过程可分为词法分析、语法分析、语义分析、中间代码生成、代码优化和目标代码生成等阶段，还需要进行出错处理和符号表管理。符号表的作用是记录源程序中各个符号的必要信息，以辅助语义的正确性检查和代码生成，在编译过程中需要对符号表进行快速有效地查找、插入、修改和删除等操作。符号表的建立可以始于词法分析阶段，也可以放到语法分析和语义分析阶段，但符号表的使用有时会延续到目标代码的运行阶段。

参考答案

（22）A

试题（23）

设计操作系统时不需要考虑的问题是__（23）__。

（23）A. 计算机系统中硬件资源的管理　B. 计算机系统中软件资源的管理

C. 用户与计算机之间的接口　D. 语言编译器的设计实现

试题（23）分析

操作系统设计的目的是管理计算机系统中的软硬件资源，为用户与计算机之间提供方便的接口。

参考答案

（23）D

试题（24）、（25）

假设某计算机系统中资源 R 的可用数为 6，系统中有 3 个进程竞争 R，且每个进程都需要 i 个 R，该系统可能会发生死锁的最小 i 值是__（24）__。若信号量 S 的当前值为 −2，则 R 的可用数和等待 R 的进程数分别为__（25）__。

（24）A. 1　B. 2　C. 3　D. 4

（25）A. 0、0　B. 0、1　C. 1、0　D. 0、2

试题（24）、（25）分析

本题考查对操作系统进程管理信号量方面的基础知识。

选项 A 是错误的，因为每个进程都需要 1 个资源 R，系统为 3 个进程各分配 1 个，系统中资源 R 的可用数为 3，3 个进程都能得到所需资源，故不发生死锁；选项 B 是错

误的，因为，每个进程都需要 2 个资源 R，系统为 3 个进程各分配 2 个，系统中资源 R 的可用数为 6，3 个进程都能得到所需资源，故也不发生死锁；选项 C 是正确的，因为，每个进程都需要 3 个资源 R，系统为 3 个进程各分配 2 个，系统中资源 R 的可用数为 6，3 个进程再申请 1 个资源 R 得不到满足，故发生死锁；选项 D 显然是错误的，分析略。

试题（25）的正确的答案为选项 D。因为，早在 1965 年荷兰学者 Dijkstra 提出信号量机制是一种有效的进程同步与互斥工具。目前，信号量机制有了很大的发展，主要有整型信号量、记录型信号量和信号量集机制。

对于整型信号量，可以根据控制对象的不同被赋予不同的值。通常将信号量分为公用信号量和私用信号量两类。其中，公用信号量用于实现进程间的互斥，初值为 1 或资源的数目；私用信号量用于实现进程间的同步，初值为 0 或某个正整数。信号量 S 的物理意义是：S≥0 表示某资源的可用数，若 S<0，则其绝对值表示阻塞队列中等待该资源的进程数。本题由于信号量 S 的当前值为–2，则意味着系统中资源 R 的可用个数 M=0，等待资源 R 的进程数 N=2。

参考答案

（24）C（25）D

试题（26）

某计算机系统页面大小为 4K，若进程的页面变换表如下所示，逻辑地址为十六进制 1D16H。该地址经过变换后，其物理地址应为十六进制 （26） 。

页号	物理块号
0	1
1	3
2	4
3	6

（26）A．1024H　　B．3D16H　　C．4D16H　　D．6D16H

试题（26）分析

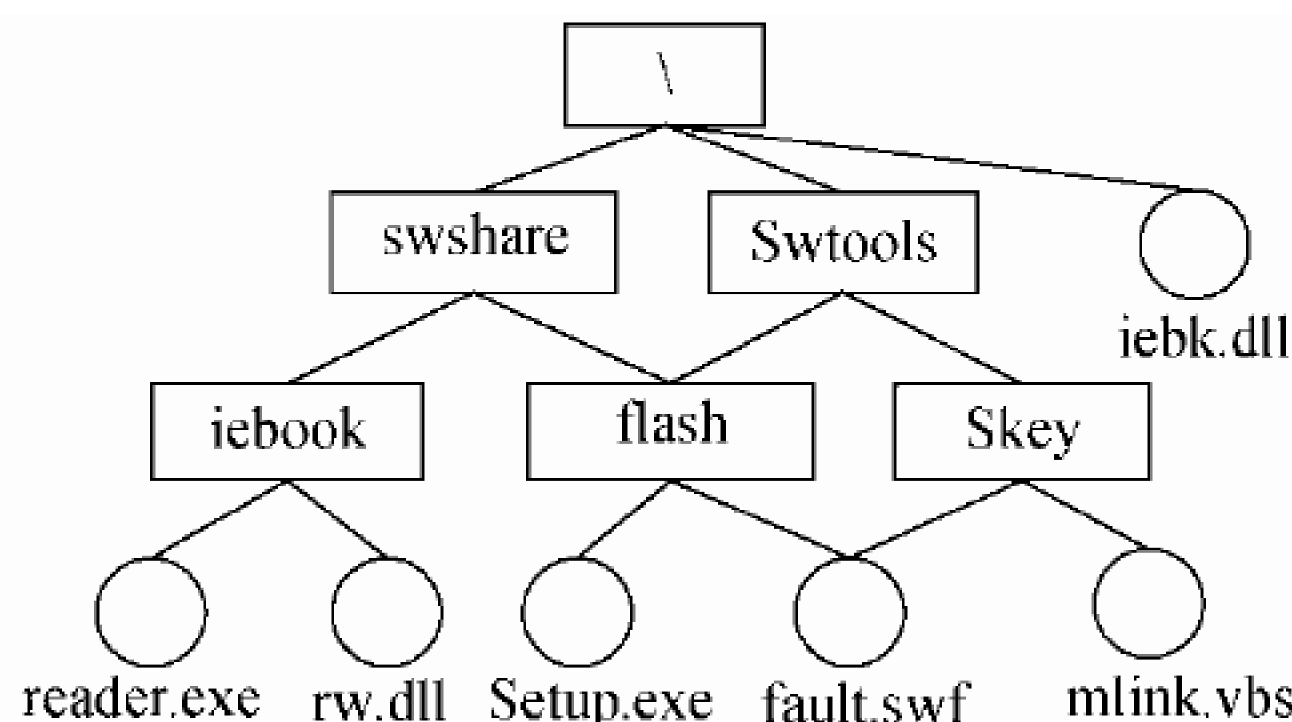
根据题意页面大小为 4K，逻辑地址为十六进制 1D16H 其页号为 1，页内地址为 D16H，查页表后可知物理块号为 3，该地址经过变换后，其物理地址应为物理块号 3 拼上页内地址 C16H，即十六进制 3D16H。

参考答案

（26）B

试题（27）、（28）

若某文件系统的目录结构如下图所示，假设用户要访问文件 fault.swf，且当前工作目录为 swshare，则该文件的全文件名为 （27） ，相对路径和绝对路径分别为 （28） 。



（27）A．fault.swf　　B．flash\fault.swf
C．swshare\flash\fault.swf　　D．\swshare\flash\fault.swf

（28）A．swshare\flash\和\flash\　　B．flash\和\swshare\flash\
C．\swshare\flash\和 flash\　　D．\flash\和\swshare\flash\

试题（27）、（28）分析

本题考查对操作系统文件管理方面的基础知识。

路径名是指操作系统查找文件所经过的目录名以及目录名之间的分隔符构成的。通常，操作系统中全文件名是指路径名+文件名。

按查找文件的起点不同可以将路径分为：绝对路径和相对路径。从根目录开始的路径称为绝对路径；从用户当前工作目录开始的路径称为相对路径，相对路径是随着当前工作目录的变化而改变的。

参考答案

（27）D　（28）B

试题（29）

以下关于统一过程 UP 的叙述中，不正确的是__（29）__。

（29）A．UP 是以用例和风险为驱动，以架构为中心，迭代并且增量的开发过程
B．UP 定义了四个阶段，即起始、精化、构建和确认阶段
C．每次迭代都包含计划、分析、设计、构造、集成、测试以及内部和外部发布
D．每个迭代有五个核心工作流

试题（29）分析

本题考查软件过程模型的基础知识。

UP（统一过程）模型是一种以用例和风险为驱动、以架构为中心、迭代并且增量的开发过程，由 UML 方法和工具支持。UP 过程定义了五个阶段，起始阶段、精化阶段、构建阶段、移交阶段和产生阶段。开发过程中有多次迭代，每次迭代都包含计划、分析、设计、构造、集成和测试，以及内部和外部发布。每个迭代有五个核心工作流，捕获系统应该做什么的需求工作流、精化和结构化需求的分析工作流、在系统结构内实现需求的设计工作流、构造软件的实现工作流和验证是否如期望那样工作的测试工作流。

参考答案

（29）B

试题（30）

某公司要开发一个软件产品，产品的某些需求是明确的，而某些需求则需要进一步细化。由于市场竞争的压力，产品需要尽快上市，则开发该软件产品最不适合采用__（30）__模型。

（30）A．瀑布　　B．原型　　C．增量　　D．螺旋

试题（30）分析

本题考查软件过程模型的基础知识。

瀑布模型将软件生存周期各个活动规定为线性顺序连接的若干阶段的模型，规定了由前至后，相互衔接的固定次序，如同瀑布流水，逐级下落。这种方法是一种理想的开发模式，缺乏灵活性，特别是无法解决软件需求不明确或不准确的问题。原型模型从初始的原型逐步演化成最终软件产品，特别适用于对软件需求缺乏准确认识的情况。螺旋将瀑布模型与快速原型模型结合起来，并且加入两种模型均忽略了的风险分析，适用于复杂的大型软件。增量是开发把软件产品作为一系列的增量构件来设计、编码、集成和测试，可以在增量开发过程中逐步理解需求。

参考答案

（30）A

试题（31）

在屏蔽软件错误的容错系统中，冗余附加技术的构成不包括__（31）__。

（31）A．关键程序和数据的冗余存储及调用

B．冗余备份程序的存储及调用

C．实现错误检测和错误恢复的程序

D．实现容错软件所需的固化程序

试题（31）分析

本题考查软件容错技术的基础知识。

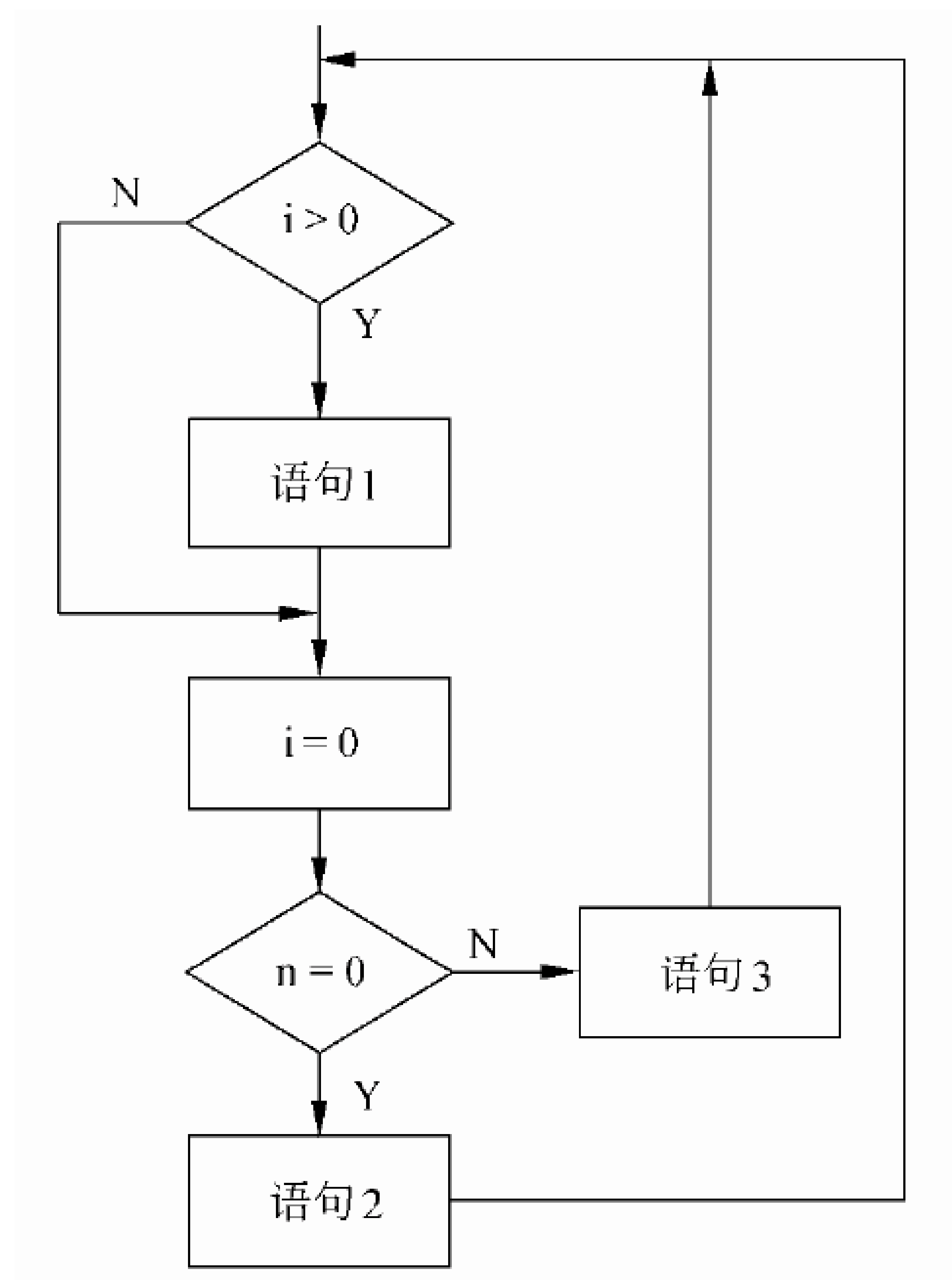
容错技术是对某些无法避开的差错，使其影响减至最小的技术。通常冗余技术分为四类，结构冗余、信息冗余、时间冗余和冗余附加技术。其中冗余附加技术是指为实现其他类型冗余技术所需要的资源和技术，包括程序指令、数据、存放和调动它们的空间和通道等。在屏蔽硬件错误的容错技术中，冗余附加技术包括：关键程序和数据的冗余存储及调用；检测、表决、切换、重构、纠错和复算的实现。在屏蔽软件错误的容错技术中，冗余附加技术包括：冗余备份程序的存储及调用；实现错误检测和错误恢复的程序；实现容错软件所需的固化程序。

参考答案

（31）A

试题（32）

采用 McCabe 度量法计算下列程序图的环路复杂性为__（32）__。



（32）A．2　　B．3　　C．4　　D．5

试题（32）分析

本题考查软件质量的基础知识。McCabe 度量法是一种基于程序控制流的复杂性度量方法，环路复杂性为 $V(G) = m - n + 2$，图中 $m = 8, n = 6$，$V(G) = 8 - 6 + 2 = 4$。

参考答案

（32）C

试题（33）

以下关于文档的叙述中，不正确的是__（33）__。

（33）A．文档仅仅描述和规定了软件的使用范围及相关的操作命令

B．文档也是软件产品的一部分，没有文档的软件就不能称之为软件

C．软件文档的编制在软件开发工作中占有突出的地位和相当大的工作量

D．高质量文档对于发挥软件产品的效益有着重要的意义

试题（33）分析

本题考查软件文档的基础知识。

软件文档不仅仅描述和规定了软件的使用范围及相关的操作命令，还包括硬件采购和网络设计中形成的文档。文档是软件产品的重要组成部分。软件文档的编制在软件开发工作中占有突出的地位和相当大的工作量，对发挥软件产品的效益具有重要的意义。

参考答案

（33）A

试题（34）

某搜索引擎在使用过程中，若要增加接受语音输入的功能，使得用户可以通过语音输入来进行搜索，此时应对系统进行__（34）__维护。

（34）A．正确性　　B．适应性　　C．完善性　　D．预防性

试题（34）分析

本题考查软件维护的基础知识。

软件维护一般包括四种类型：

正确性维护是指改正在系统开发阶段已发生而系统测试阶段尚未发现的错误；

适应性维护是指使应用软件适应新型技术变化和管理需求变化而进行的修改；

完善性维护是指为扩充功能和改善性能而进行的修改，主要是指对已有的软件系统增加一些在系统分析和设计阶段中没有规定的功能与性能特征；

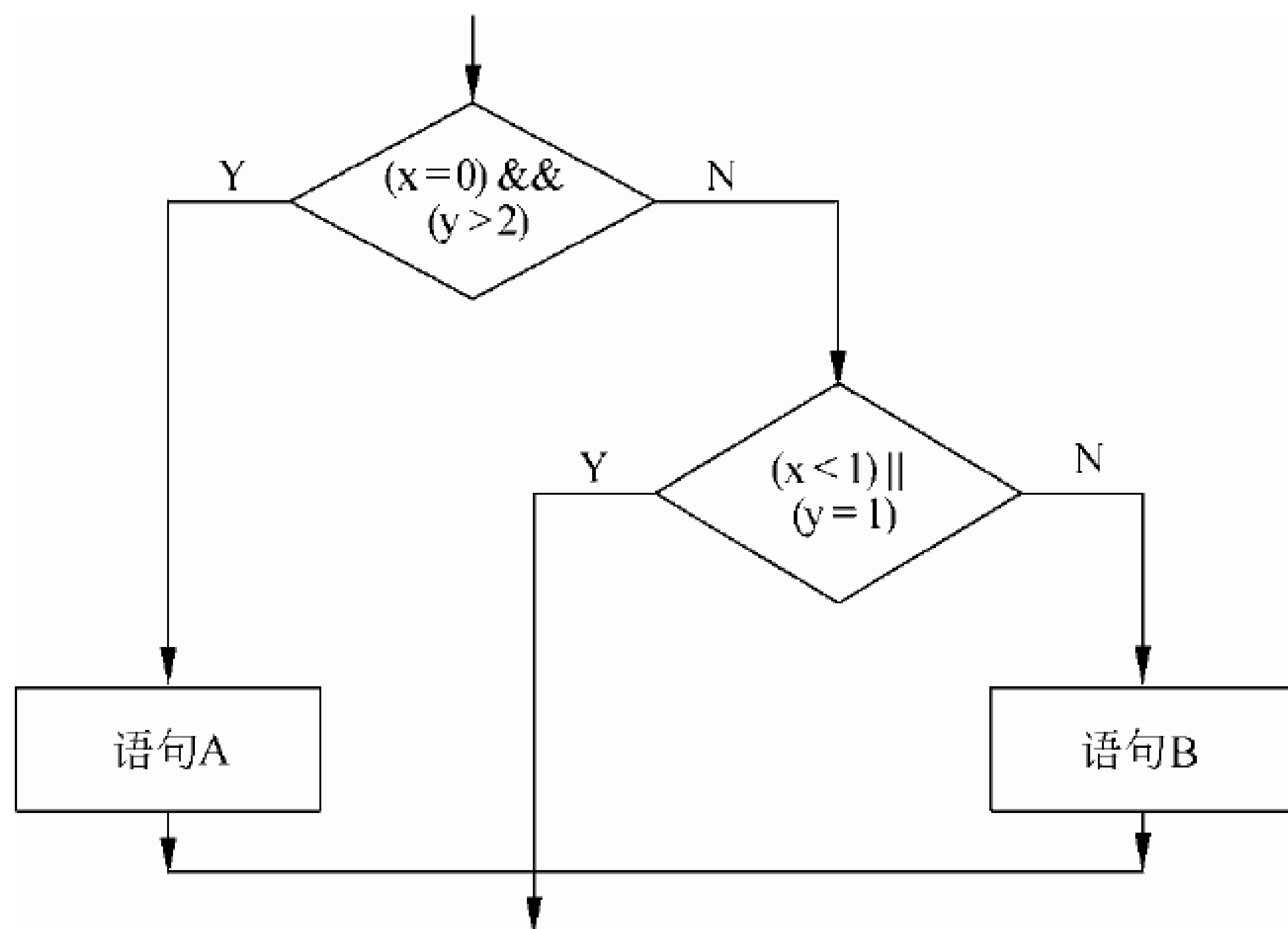
预防性维护是指为了改进应用软件的可靠性和可维护性，为了适应未来的软硬件环境的变化，主动增加预防性的功能，以使应用系统适应各类变化而不被淘汰。

参考答案

（34）B

试题（35）、（36）

采用白盒测试方法对下图进行测试，设计了 4 个测试用例：①（x=0, y=3），②（x=1, y=2），③（x=-1, y=2），④（x=3, y=1）。至少需要测试用例①②才能完成__（35）__覆盖，至少需要测试用例①②③或①②④才能完成__（36）__覆盖。



（35）A．语句　　B．条件　　C．判定/条件　　D．路径

（36）A．语句　　B．条件　　C．判定/条件　　D．路径

试题（35）、（36）分析

本题考查软件测试的基础知识。

测试用例①（x=0, y=3）在第一个判断结果为 Y，执行语句 A；测试用例②（x=1, y=2）在第一个判断结果为 N，第二个判断结果为 N，执行语句 B；测试用例③（x=-1, y=2）和④（x=3, y=1）在第一个判断结果为 N，第二个判断结果为 Y。至少需要测试用例①②才能完成语句覆盖，至少需要测试用例①②③或①②④才能完成路径覆盖。

参考答案

（35）A　　（36）D

试题（37）

<u>（37）</u>是一个类与它的一个或多个细化类之间的关系，即一般与特殊的关系。

（37）A．泛化　　B．关联　　C．聚集　　D．组合

试题（37）分析

本题考查面向对象的基本知识。

泛化是一个类与它的一个或多个细化类之间的关系，表达一般与特殊的关系。关联是类与类之间的一种结构关系。聚集是一种关系，其中一个较大的整体类包含一个或多个较小的部分类；相反地，一个较小的部分类是一个较大的整体类的一部分。组合是一种聚合关系，其中整体负责其部分的创建和销毁，如果整体不存在了，部分也将不存在。

参考答案

（37）A

试题（38）

某些程序设计语言中，在运行过程中当一个对象发送消息请求服务时，根据接收对象的具体情况将请求的操作与实现的方法进行连接，称为<u>（38）</u>。

（38）A．静态绑定　　B．通用绑定　　C．动态绑定　　D．过载绑定

试题（38）分析

本题考查面向对象的基本知识。

绑定是一个把过程调用和响应调用所需要执行的代码加以结合的过程。在一般的程序设计语言中，绑定是在编译时进行的，叫做静态绑定。动态绑定则是在运行时进行的，因此，一个给定的过程调用和代码的结合直到调用发生时才进行。

参考答案

（38）C

试题（39）、（40）

在面向对象技术中，不同的对象在收到同一消息时可以产生完全不同的结果，这一现象称为<u>（39）</u>，它由<u>（40）</u>机制来支持。利用类的层次关系，把具有通用功能的消息存放在高层次，而不同的实现这一功能的行为放在较低层次，在这些低层次上生成的对象能够给通用消息以不同的响应。

（39）A. 绑定　　B. 继承　　C. 消息　　D. 多态

（40）A. 绑定　　B. 继承　　C. 消息　　D. 多态

试题（39）、（40）分析

本题考查面向对象的基本知识。

面向对象技术中，继承关系是一种模仿现实世界中继承关系的一种类之间的关系，是超类（父类）和子类之间共享数据和方法的机制。父类定义公共的属性和操作，一个父类可以有多个子类，即多个特例。子类可以继承其父类或祖先类中的属性和操作作为自己的内容而不必自己定义，也可以覆盖这些操作，并加入新的内容。绑定是一个把过程调用和响应调用所需要执行的代码加以结合的过程。绑定有在编译时进行的，即静态绑定，有在运行时进行的，即动态绑定。不同的对象收到同一消息可以进行不同的响应，产生完全不同的结果，用户可以发送一个通用的消息，而实现细节则由接收对象自行决定，使得同一个消息就可以调用不同的方法，即一个对象具有多种形态，称为多态。不同类的对象通过消息相互通信。

参考答案

（39）D　　（40）B

试题（41）

对一个复杂用例中的业务处理流程进行进一步建模的最佳工具是 UML__（41）__。

（41）A. 状态图　　B. 顺序图　　C. 类图　　D. 活动图

试题（41）分析

本题考查采用统一建模语言（UML）的基本知识。

采用 UML 对系统进行建模时，首先确定系统边界，识别出主要用例，建模用例图。然后对用例图中的复杂用例采用活动图进一步进行建模，以对用例中执行过程中对象如何通过消息相互交互进行建模。系统的领域模型采用类图进行建模，交互关系采用顺序图、交互概览图等进行建模。

参考答案

（41）D

试题（42）、（43）

如下所示的 UML 序列图中，__（42）__表示返回消息，Account 类必须实现的方法有__（43）__。

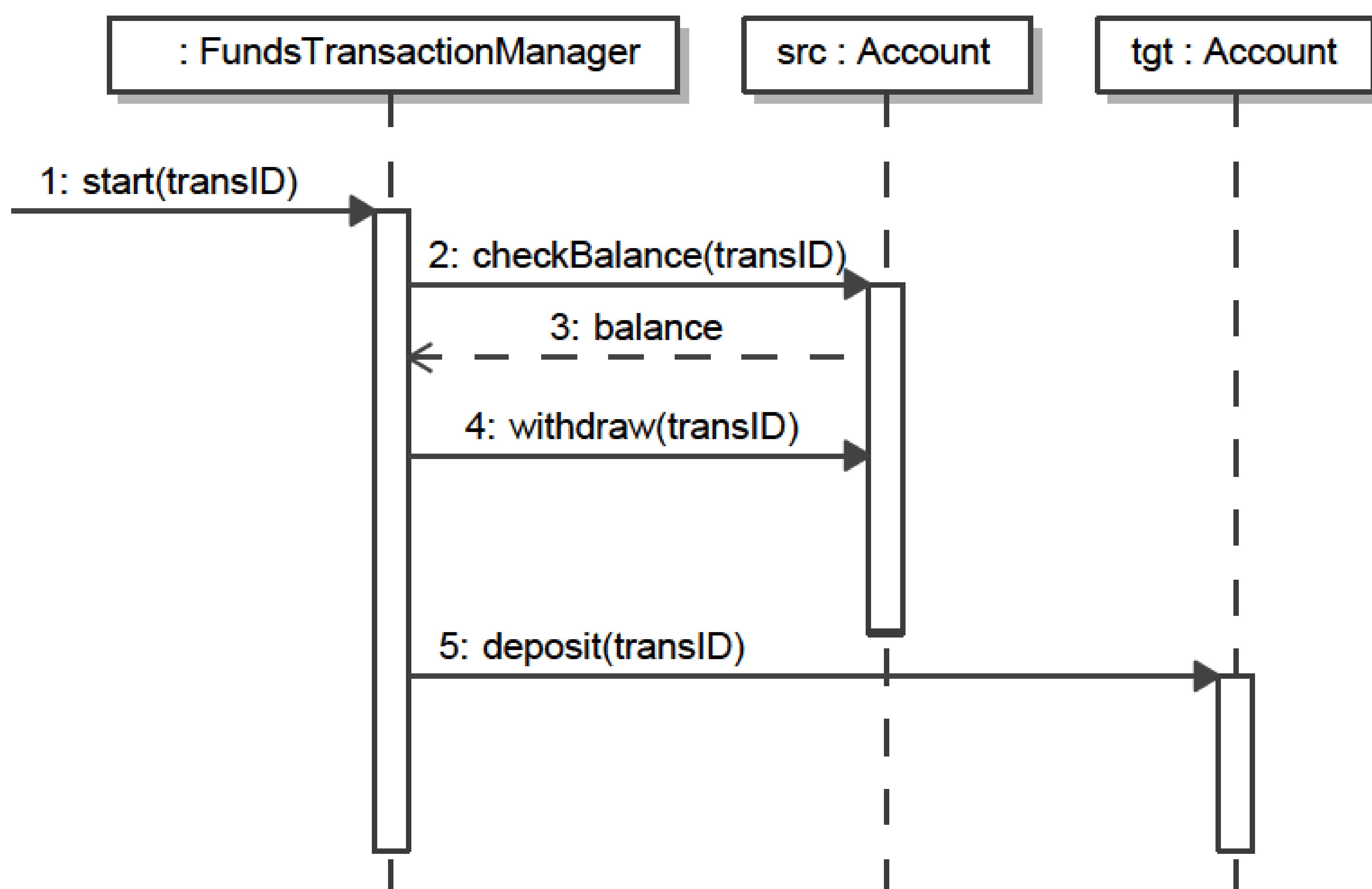
（42）A. tansID　　B. balance　　C. withdraw　　D. deposit

（43）A. start()

B. checkBalance()和 withdraw()

C. deposit()

D. checkBalance()、withdraw() 和 deposit()



试题（42）、（43）分析

本题考查采用统一建模语言（UML）的基本知识。

UML 序列图（Sequence Diagram）以二维图的形式显示对象之间交互，纵轴自上而下表示时间，横轴表示要交互的对象，主要体现对象间消息传递的时间顺序，强调参与交互的对象及其间消息交互的时序。序列图中包括的建模元素主要有：活动者（Actor）、对象（Object）、生命线（Lifeline）、控制焦点（Focus of control）和消息（Message）等。其中对象名标有下划线；生命线表示为虚线，沿竖线向下延伸；消息在序列图中标记为箭头；控制焦点由薄矩形表示。

消息是从一个对象的生命线到另一个对象生命线的箭头，用从上而下的时间顺序来安排。一般分为同步消息（ ⟶ ），异步消息（ ⟶ ）和返回消息（ ⟵ -- ）。本题图中 balance 为返回消息，其他为同步消息。src 和 tgt 均为 Account 对象，所以 Account 应该实现 checkBalance()、withdraw() 和 deposit()方法，FundsTransactionManager 应该实现 start()方法。

参考答案

（42）B　　　（43）D

试题（44）、（45）

下图所示为__（44）__设计模式，适用于__（45）__。

（44）A．适配器（Adapter）　　　B．责任链（Chain of Responsibility）

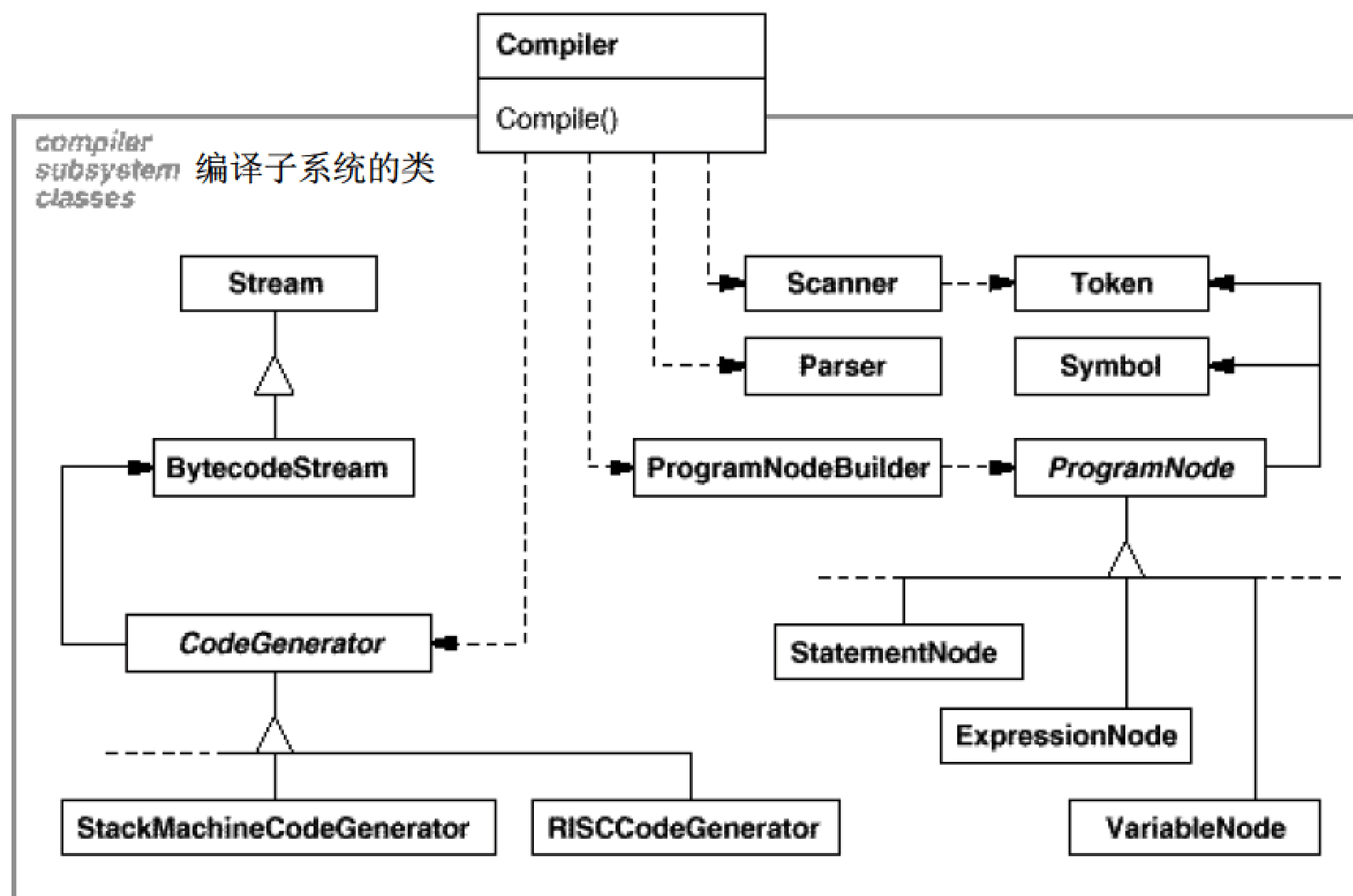
C．外观（Facade）　　　D．桥接（Bridge）

（45）A．有多个对象可以处理一个请求，在运行时刻自动确定由哪个对象处理

B．想使用一个已经存在的类，而其接口不符合要求

C．类的抽象和其实现之间不希望有一个固定的绑定关系

D．需要为一个复杂子系统提供一个简单接口



试题（44）、（45）分析

本题考查设计模式的基本概念。

每种设计模式都有特定的意图和适用场景。

适配器（Adapter）模式将一个类的接口转换成客户希望的另外一个接口，使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。适用于想使用一个已经存在的类，而其接口不符合要求的情况。

责任链（Chain of Responsibility）模式使多个对象都有机会处理请求，从而避免请求的发送者和接收者之间的耦合关系，将这些对象连成一条链，并沿着这条链传递该请求，直到有一个对象处理它为止。适用于有多个的对象可以处理一个请求，哪个对象处理该请求运行时刻自动确定的情况。

桥接（Bridge）模式将抽象部分与其实现部分分离，使它们都可以独立地变化。适用于不希望在抽象和它的实现部分之间有一个固定的绑定关系的情况。

外观（Facade）模式为子系统中的一组接口提供一个一致的界面，Facade 模式定义了一个高层接口，这个接口使得这一子系统更加容易使用。适用于需要为一个复杂子系统提供一个简单接口的情况。

参考答案

（44）C （45）D

试题（46）、（47）

下列设计模式中，__（46）__模式既是类结构型模式，又是对象结构型模式。此模式与__（47）__模式类似的特征是，都给另一个对象提供了一定程度上的间接性，都涉及从自身以外的一个接口向这个对象转发请求。

（46）A．桥接（Bridge）　B．适配器（Adapter）
C．组成（Composite）　D．装饰器（Decorator）
（47）A．桥接（Bridge）　B．适配器（Adapter）
C．组成（Composite）　D．装饰器（Decorator）

试题（44）、（47）分析

本题考查设计模式的基本概念。

每种设计模式都有特定的意图，描述一个在我们周围不断重复发生的问题，以及该问题的解决方案的核心，使该方案能够重用而不必做重复劳动。

适配器（Adapter）模式将一个类或对象的接口转换成客户希望的另外一个接口，使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。既是类结构模式，又是对象结构模式。桥接（Bridge）模式将抽象部分与其实现部分分离，使它们都可以独立地变化。适配器模式和桥接模式具有类似的特征，都给另一个对象提供了一定程度上的间接性，都涉及自身以外的一个接口向这个对象转发请求。

组合（Composite）模式将对象组合成树形结构以表示“部分-整体”的层次结构，使得用户对单个对象和组合对象的使用具有一致性。装饰器（Decorator）模式描述了以透明围栏来支持修饰的类和对象的关系，动态地给一个对象添加一些额外的职责，从增加功能的角度来看，装饰器模式相比生成子类更加灵活。

参考答案

（46）B　（47）A

试题（48）

以下关于实现高级程序设计语言的编译和解释方式的叙述中，正确的是__（48）__。

（48）A．在编译方式下产生源程序的目标程序，在解释方式下不产生
B．在解释方式下产生源程序的目标程序，在编译方式下不产生
C．编译和解释方式都产生源程序的目标程序，差别是优化效率不同
D．编译和解释方式都不产生源程序的目标程序，差别在是否优化

试题（48）分析

本题考查程序语言基础知识。

用某种高级语言或汇编语言编写的程序称为源程序，源程序不能直接在计算机上执行。如果源程序是用汇编语言编写的，则需要一个称为汇编程序的翻译程序将其翻译成目标程序后才能执行。如果源程序是用某种高级语言编写的，则需要对应的解释程序或编译程序对其进行翻译，然后在机器上运行。

解释程序也称为解释器，它可以直接解释执行源程序，或者将源程序翻译成某种中间表示形式后再加以执行；而编译程序（编译器）则首先将源程序翻译成目标语言程序，然后在计算机上运行目标程序。这两种语言处理程序的根本区别是：在编译方式下，机器上运行的是与源程序等价的目标程序，源程序和编译程序都不再参与目标程序的执行

过程；而在解释方式下，解释程序和源程序（或其某种等价表示）要参与到程序的运行过程中，运行程序的控制权在解释程序。解释器翻译源程序时不产生独立的目标程序，而编译器则需将源程序翻译成独立的目标程序。

参考答案

（48）A

试题（49）

大多数程序设计语言的语法规则用__（49）__描述即可。

（49）A．正规文法　　B．上下文无关文法
　　C．上下文有关文法　　D．短语结构文法

试题（49）分析

本题考查程序语言基础知识。

乔姆斯基文法体系共分为短语结构文法、上下文有关文法、上下文有关文法和正规文法4类。

短语结构文法或无限制文法也称为0型文法，其描述能力相当于图灵机，可使用任何的语法描述形式。

上下文有关文法也称为1型文法，其描述能力相当于线性有界自动机，语法形式为：xSy -> xAy。也就是说，S（非终结符号）推导出A（非终结符号与终结符号的混合串）是和上下文x, y相关的，即S只有在上下文x, y的环境中才能推导出A。

上下文无关文法也称为2型文法，其描述能力相当于下推自动机，语法形式如下：S -> A，即S可以无条件的推导出A，与上下文无关。

正规文法也称为3型文法，等价于正则表达式，其描述能力相当于有穷自动机，语法形式如下：S -> Aa，其中最后一个a必须为非终结符。

大多数程序语言的语法现象可用上下文无关文法描述。

参考答案

（49）B

试题（50）

在某C/C++程序中，整型变量a的值为0且应用在表达式“c=b/a”中，则最可能发生的情形是__（50）__。

（50）A．编译时报告有语法错误　　B．编译时报告有逻辑错误
　　C．运行时报告有语法错误　　D．运行时产生异常

试题（50）分析

本题考查程序语言基础知识。

对程序中含有变量的表达式求值发生在运行时，若除数为0进行除运算在运行时报告异常。

参考答案

（50）D

试题（51）、（52）

为了保证数据库中数据的安全可靠和正确有效，系统在进行事务处理时，对数据的插入、删除或修改的全部有关内容先写入 （51） ；当系统正常运行时，按一定的时间间隔，把数据库缓冲区内容写入 （52） ；当发生故障时，根据现场数据内容及相关文件来恢复系统的状态。

（51）A．索引文件　B．数据文件　C．日志文件　D．数据字典

（52）A．索引文件　B．数据文件　C．日志文件　D．数据字典

试题（51）、（52）分析

本题考查关系数据库事务处理方面的基础知识。

为了保证数据库中数据的安全可靠和正确有效，数据库管理系统（DBMS）提供数据库恢复、并发控制、数据完整性保护与数据安全性保护等功能。数据库在运行过程中由于软硬件故障可能造成数据被破坏，数据库恢复就是在尽可能短的时间内，把数据库恢复到故障发生前的状态。具体的实现方法有多种，如：定期将数据库作备份；在进行事务处理时，将数据更新（插入、删除、修改）的全部有关内容写入日志文件；当系统正常运行时，按一定的时间间隔，设立检查点文件，把内存缓冲区内容还未写入到磁盘中去的有关状态记录到检查点文件中；当发生故障时，根据现场数据内容、日志文件的故障前映像和检查点文件来恢复系统的状态。

参考答案

（51）C　（52）B

试题（53）

“当多个事务并发执行时，任一事务的更新操作直到其成功提交的整个过程，对其他事务都是不可见的”，这一性质通常被称为事务的 （53） 。

（53）A．原子性　B．一致性　C．隔离性　D．持久性

试题（53）分析

本题考查数据库事务处理方面的知识。

事务具有原子性（atomicity）、一致性（consistency）、隔离性（isolation）和持久性（durability）。这 4 个特性也称事务的 ACID 性质。其中，事务的隔离性是指事务相互隔离，即当多个事务并发执行时，任一事务的更新操作直到其成功提交的整个过程，对其他事务都是不可见的。

参考答案

（53）C

试题（54）～（56）

假定某企业 2014 年 5 月的员工工资如下表所示：

2014年5月员工工资表

员工号	姓名	部门	基本工资	岗位工资	全勤奖	应发工资	扣款	实发工资
1001	王小龙	办公室	680.00	1200.00	100.00	1980.00	20.00	1960.00
1002	孙晓红	办公室	1200.00	1000.00	0.00	2200.00	50.00	2150.00
2001	赵眙珊	企划部	680.00	1200.00	100.00	1980.00	10.00	1970.00
2002	李丽敏	企划部	950.00	2000.00	100.00	3050.00	15.00	3035.00
3002	傅学君	设计部	800.00	1800.00	0.00	2600.00	50.00	2550.00
3003	曹海军	设计部	950.00	1600.00	100.00	2650.00	20.00	2630.00
3004	赵晓勇	设计部	1200.00	2500.00	0.00	3700.00	50.00	3650.00
4001	杨一凡	销售部	680.00	1000.00	100.00	1780.00	10.00	1770.00
4003	景昊星	销售部	1200.00	2200.00	100.00	3500.00	20.00	3480.00
4005	李建军	销售部	850.00	1800.00	100.00	2750.00	98.00	2652.00

查询人数大于2的部门和部门员工应发工资的平均工资的SQL语句如下：

```
SELECT____(54)____
FROM  工资表
____(55)____
____(56)____;
```

（54）A．部门，AVG（应发工资）AS 平均工资

B．姓名，AVG（应发工资）AS 平均工资

C．部门，平均工资 AS AVG（应发工资）

D．姓名，平均工资 AS AVG（应发工资）

（55）A．ORDER BY 姓名　　B．ORDER BY 部门

C．GROUP BY 姓名　　D．GROUP BY 部门

（56）A．WHERE COUNT(姓名)> 2

B．WHERE COUNT(DISTINCT(部门))> 2

C．HAVING COUNT(姓名)> 2

D．HAVING COUNT(DISTINCT(部门))> 2

试题（54）～（56）分析

本题考查SQL方面的基础知识。

查询各部门人数大于2和部门员工的平均工资的SQL语句如下：

```
SELECT 部门，AVG（应发工资）AS 平均工资
FROM  工资表
GROUP BY 部门
HAVING COUNT(姓名)> 2;
```

SQL 提供可为关系和属性重新命名的机制，这是通过使用具有“Old-name as

new-name”形式的 as 子句来实现的。As 子句即可出现在 select 子句，也可出现在 from 子句中。

试题（55）的正确的答案为选项 D。因为本题是按部门进行分组，ORDER BY 子句的含义是对其后跟着的属性进行排序，故选项 A 和 B 均是错误的；GROUP BY 子句就是对元组进行分组，保留字 GROUP BY 后面跟着一个分组属性列表。根据题意，要查询部门员工的平均工资，选项 C 显然是错误的。

试题（56）的正确的答案为选项 C。因为 WHERE 语句是对表进行条件限定，所以选项 A 和 B 均是错误的。在 GROUP BY 子句后面跟一个 HAVING 子句可以对元组在分组前按照某种方式加上限制。COUNT（*）是某个关系中所有元组数目之和，但 COUNT（A）却是 A 属性非空的元组个数之和。COUNT(DISTINCT(部门))的含义是对部门属性值相同的只统计 1 次。HAVING COUNT(DISTINCT(部门))语句分类统计的结果均为 1，故选项 D 是错误的；HAVING COUNT(姓名) 语句是分类统计各部门员工，故正确的答案为选项 C。

参考答案

（54）A　（55）D　（56）C

试题（57）

若对线性表的最常用操作是访问任意指定序号的元素，并在表尾加入和删除元素，则适宜采用 <u>（57）</u> 存储。

（57）A．顺序表　　　　B．单链表

C．双向链表　　　　D．哈希表

试题（57）分析

本题考查数据结构基础知识。

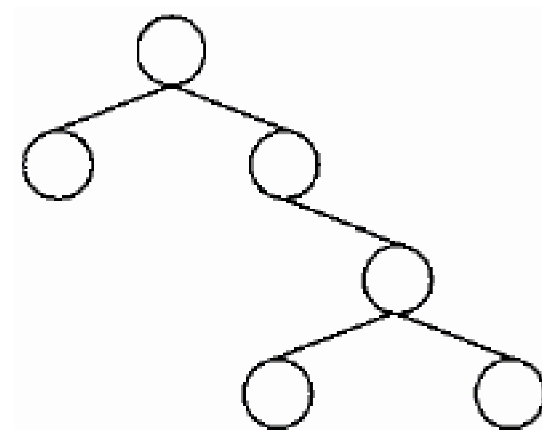
线性表的元素在逻辑上是一个线性序列，若最常用的操作是访问任意指定序号的元素，而且其插入和删除元素的操作均在表尾进行，不需要移动其他元素，则其存储结构采用顺序表最为合适。

参考答案

（57）A

试题（58）、（59）

某二叉树如右图所示，若进行顺序存储（即用一维数组元素存储该二叉树中的结点且通过下标反映结点间的关系，例如，对于下标为 i 的结点，其左孩子的下标为 2i、右孩子的下标为 2i+1），则该数组的大小至少为 <u>（58）</u>；若采用三叉链表存储该二叉树（各个结点包括结点的数据、父结点指针、左孩子指针、右孩子指针），则该链表的所有结点中空指针的数目为 <u>（59）</u>。



（58）A．6　　B．10　　C．12　　D．15

（59）A．6　　B．8　　C．12　　D．14

试题（58）、（59）分析

本题考查数据结构基础知识。

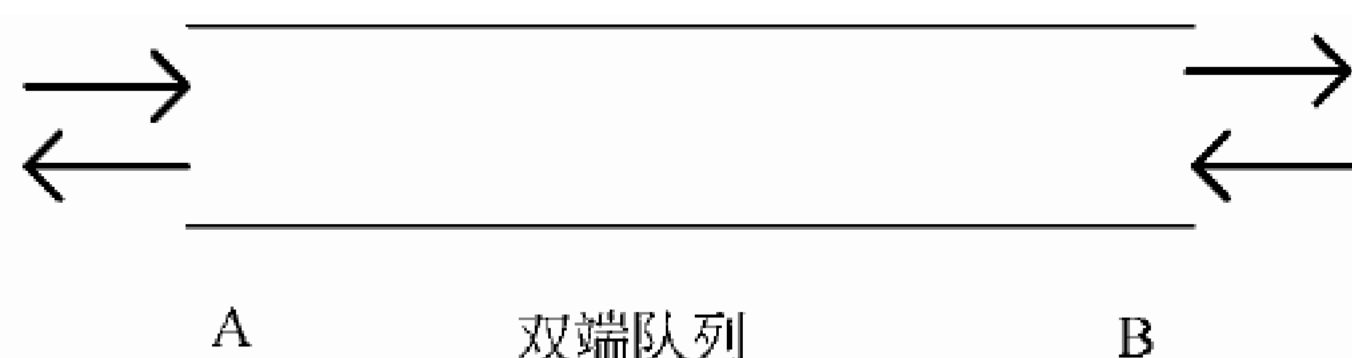
题图所示的二叉树有 6 个结点，根结点的编号为 1，其左孩子和右孩子分别为 2 和 3，按照右孩子链继续，3 号结点的右孩子编号为 7，7 号结点的右孩子编号为 15，因此该二叉树进行顺序存储时数组大小至少为 15。采用三叉链表存储时，每个结点有 3 个指针域，共 18 个指针域，其中，12 个孩子指针用了 5 个，剩余 7 个为空指针，6 个父结点指针用了 5 个，剩余 1 个为空（即根结点无双亲），因此，结点中的指针域有 8 个为空。

参考答案

（58）D　　（59）B

试题（60）

某双端队列如下图所示，要求元素进出队列必须在同一端口，即从 A 端进入的元素必须从 A 端出、从 B 端进入的元素必须从 B 端出，则对于 4 个元素的序列 e1、e2、e3、e4，若要求前 2 个元素（e1、e2）从 A 端口按次序全部进入队列，后两个元素（e3、e4）从 B 端口按次序全部进入队列，则可能得到的出队序列是__（60）__。

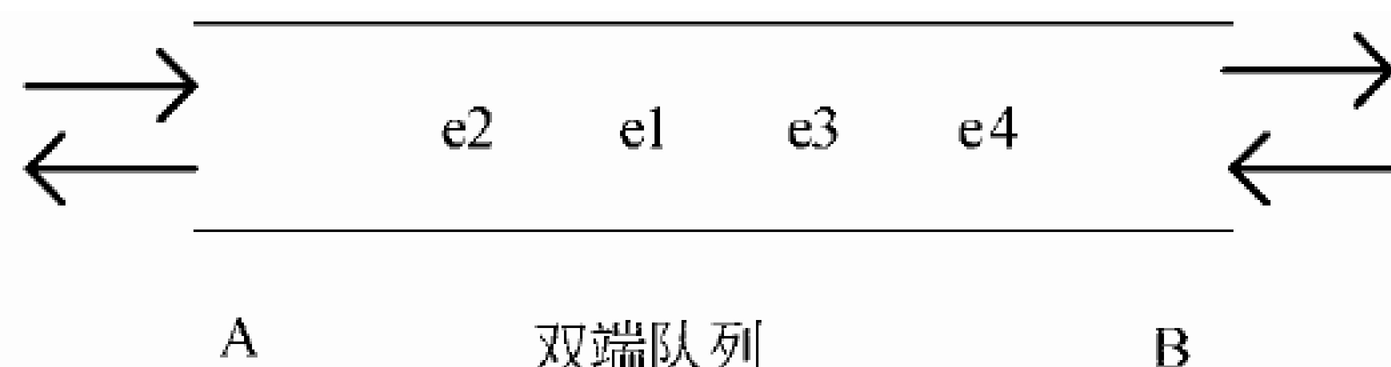


（60）A．e1、e2、e3、e4　　B．e2、e3、e4、e1
　　C．e3、e4、e1、e2　　D．e4、e3、e2、e1

试题（60）分析

本题考查数据结构基础知识。

按照题目所述，当 e1、e2 从 A 端口按次序全部进入队列，e3、e4 从 B 端口按次序全部进入队列后，双端队列的状态如下图所示。



在这种情形下，e1 和 e3 不可能先出队列，所以排除选项 A 和 C。若 e2 先出队列，则剩下的 3 个元素中，只能是 e1 或 e4 出队列，所以 e2、e3、e4、e1 是不可能的出队序列，这样就排除了选项 B。选项 D 的 e4、e3、e2、e1 是可能的出队序列。

参考答案

（60）D

试题（61）

实现二分查找（折半查找）时，要求查找表__（61）__。

（61）A．顺序存储，关键码无序排列

B．顺序存储，关键码有序排列

C．双向链表存储，关键码无序排列

D．双向链表存储，关键码有序排列

试题（61）分析

本题考查数据结构基础知识。

二分查找是一种高效的查找方法，其思路是待查找元素先与序列中间位置上的元素比较，若相等，则查找成功；若待查找元素较大，则接下来到序列的后半区进行二分查找，否则到序列的前半区进行二分查找。显然，要快速定位序列的中间位置，查找表必须进行顺序存储；其次，从二分查找过程可知，序列必须有序排列才行。

参考答案

（61）B

试题（62）、（63）

某个算法的时间复杂度递归式 T(n)=T(n-1)+n，其中 n 为问题的规模，则该算法的渐进时间复杂度为__（62）__，若问题的规模增加了 16 倍，则运行时间增加__（63）__倍。

（62）A．$\Theta(n)$　　B．$\Theta(n\lg n)$　　C．$\Theta(n^2)$　　D．$\Theta(n^2\lg n)$

（63）A．16　　B．64　　C．256　　D．1024

试题（62）、（63）分析

本题考查算法分析的基础知识。

直接展开递归式 T(n)=T(n-1)+n

$$
\begin{aligned}
&= T(n-2) + (n-1) + n \\
&= T(n-3) + (n-2) + (n-1) + n \\
&= 1 + 2 + \ldots + n \\
&= n(n-1)/2 \\
&= \Theta(n^2)。
\end{aligned}
$$

得到该算法的时间复杂度为 $\Theta(n^2)$，当问题的规模增加了 16 倍时，运行时间增加了 16^2=256 倍。

参考答案

（62）C　　（63）C

试题（64）、（65）

Prim 算法和 Kruscal 算法都是无向连通网的最小生成树的算法，Prim 算法从一个顶点开始，每次从剩余的顶点中加入一个顶点，该顶点与当前的生成树中的顶点的连边权重最小，直到得到一颗最小生成树；Kruscal 算法从权重最小的边开始，每次从不在当前

的生成树顶点之间的边中选择权重最小的边加入，直到得到一颗最小生成树，这两个算法都采用了__（64）__设计策略，且__（65）__。

（64）A．分治　　B．贪心　　C．动态规划　　D．回溯

（65）A．若网较稠密，则 Prim 算法更好

B．两个算法得到的最小生成树是一样的

C．Prim 算法比 Kruscal 算法效率更高

D．Kruscal 算法比 Prim 算法效率更高

试题（64）、（65）分析

本题考查算法设计与分析的基础知识。

Prim 算法从扩展顶点开始，每次总是“贪心的”选择与当前顶点集合中距离最短的顶点，而 Kruscal 算法从扩展边开始，每次总是“贪心的”选择剩余的边中最小权重的边，因此两个算法都是基于贪心策略进行的。

Prim 算法的时间复杂度为 $O(n^2)$，其中 n 为图的顶点数，该算法的计算时间与图中的边数无关，因此该算法适合于求边稠密的图的最小生成树；Kruscal 算法的时间复杂度为 O(mlgm)，其中 m 为图的边数，该算法的计算时间与图中的顶点数无关，因此该算法适合于求边稀疏的图的最小生成树。当图稠密时，用 Prim 算法效率更高。但若事先没有关于图的拓扑特征信息时，无法判断两者的优劣。由于一个图的最小生成树可能有多棵，因此不能保证用这两种算法得到的是同一棵最小生成树。

参考答案

（64）B　　（65）A

试题（66）、（67）

IP 地址块 155.32.80.192/26 包含了__（66）__个主机地址，以下 IP 地址中，不属于这个网络的地址是__（67）__。

（66）A．15　　B．32　　C．62　　D．64

（67）A．155.32.80.202　　B．155.32.80.195

C．155.32.80.253　　D．155.32.80.191

试题（66）、（67）分析

地址块 155.32.80.192/26 包含了 6 位主机地址，所以包含的主机地址为 62 个。

网络地址 155.32.80.192/26 的二进制为：　**10011011 00100000 01010000 11**000000

地址 155.32.80.202 的二进制为：　**10011011 00100000 01010000 11**001010

地址 155.32.80.191 的二进制为：　**10011011 00100000 01010000 10**111111

地址 155.32.80.253 的二进制为：　**10011011 00100000 01010000 11**111101

地址 155.32.80.195 的二进制为：　**10011011 00100000 01010000 11**000011

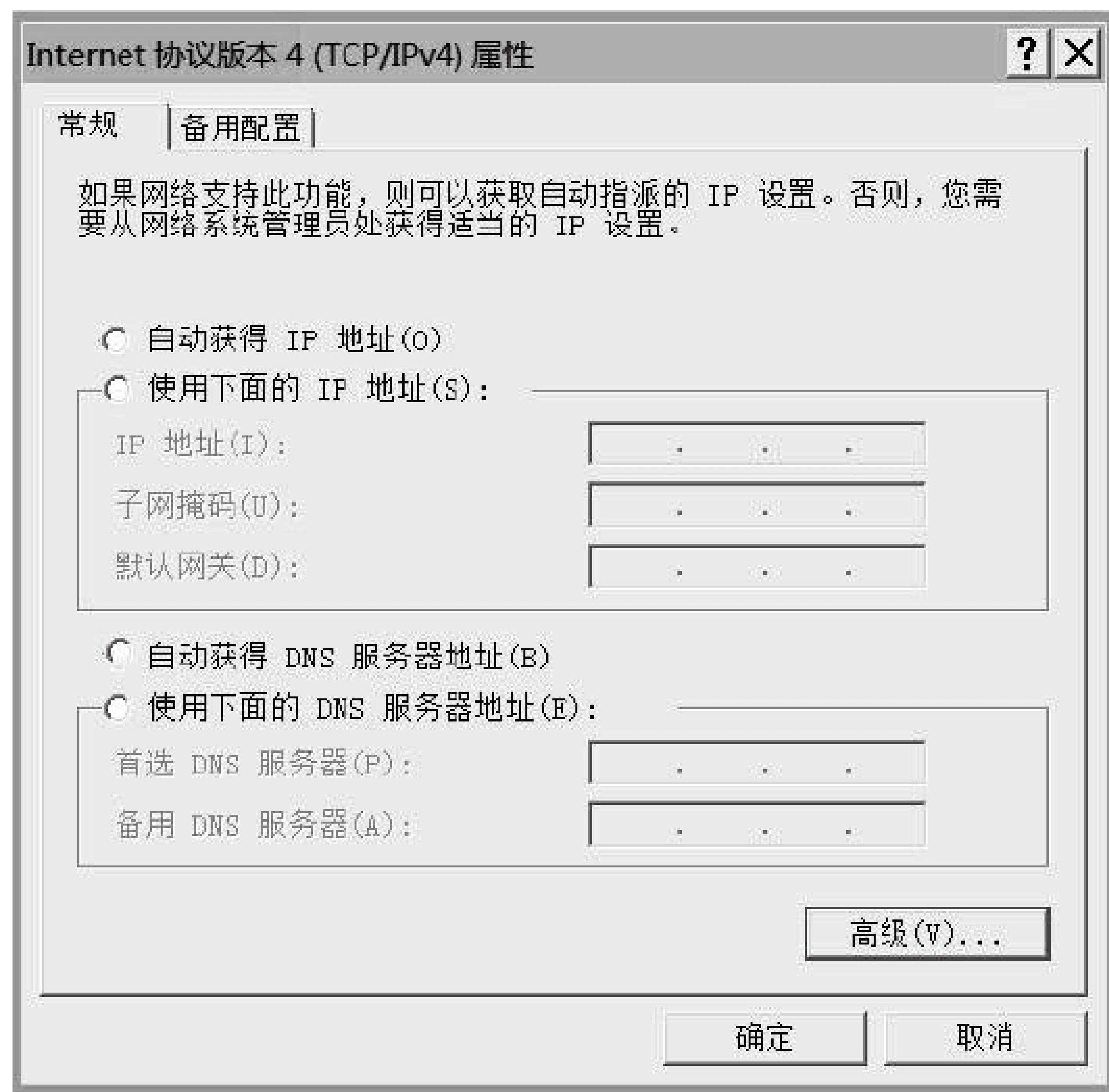
可以看出，地址 155.32.80.191 不属于网络 155.32.80.192/26。

参考答案

（66）C　（67）D

试题（68）

校园网连接运营商的 IP 地址为 202.117.113.3/30，本地网关的地址为 192.168.1.254/24，如果本地计算机采用动态地址分配，在下图中应如何配置？__（68）__。



（68）A．选取“自动获得 IP 地址”

　　B．配置本地计算机 IP 地址为 192.168.1.×

　　C．配置本地计算机 IP 地址为 202.115.113.×

　　D．在网络 169.254.×.×中选取一个不冲突的 IP 地址

试题（68）分析

如果采用动态地址分配方案，本地计算机应设置为“自动获得 IP 地址”。

参考答案

（68）A

试题（69）分析

某用户在使用校园网中的一台计算机访问某网站时，发现使用域名不能访问该网站，但是使用该网站的 IP 地址可以访问该网站，造成该故障产生的原因有很多，其中不包括__（69）__。

（69）A．该计算机设置的本地 DNS 服务器工作不正常

　　B．该计算机的 DNS 服务器设置错误

　　C．该计算机与 DNS 服务器不在同一子网

　　D．本地 DNS 服务器网络连接中断

试题（69）分析

本题主要考查网络故障判断的相关知识。

如果本地的 DNS 服务器工作不正常或者本地 DNS 服务器网络连接中断都有可能导致该计算机的 DNS 无法解析域名，而如果直接将该计算机的 DNS 服务器设置错误也会导致 DNS 无法解析域名，从而出现使用域名不能访问该网站，但是使用该网站的 IP 地址可以访问该网站。但是该计算机与 DNS 服务器不在同一子网不会导致 DNS 无法解析域名的现象发生，通常情况下大型网络里面的上网计算机与 DNS 服务器本身就不在一个子网，只要路由可达 DNS 都可以正常工作。

参考答案

（69）C

试题（70）

中国自主研发的 3G 通信标准是__（70）__。

（70）A．CDMA2000　　B．TD-SCDMA

C．WCDMA　　D．WiMAX

试题（70）分析

1985 年，ITU 提出了对第三代移动通信标准的需求，1996 年正式命名为 IMT-2000（International Mobile Telecommunications-2000），其中的 2000 有 3 层含义：

- 使用的频段在 2000MHz 附近。
- 通信速率约为 2000kb/s（即 2Mb/s）。
- 预期在 2000 年推广商用。

1999 年 ITU 批准了五个 IMT-2000 的无线电接口，这五个标准是：

- IMT-DS(Direct Spread)：即W-CDMA，属于频分双工模式，在日本和欧洲制定的UMTS系统中使用。
- IMT-MC(Multi-Carrier)：即CDMA-2000，属于频分双工模式，是第二代CDMA系统的继承者。
- IMT-TC(Time-Code)：这一标准是中国提出的TD-SCDMA，属于时分双工模式。
- IMT-SC(Single Carrier)：也称为EDGE，是一种2.75G技术。
- IMT-FT(Frequency Time)：也称为DECT。

2007 年 10 月 19 日，ITU 会议批准移动WiMAX作为第 6 个 3G 标准，称为 IMT-2000 OFDMA TDD WMAN，即无线城域网技术。

第三代数字蜂窝通信系统提供第二代蜂窝通信系统提供的所有业务类型，并支持移动多媒体业务。在高速车辆行驶时支持 144kb/s 的数据速率，步行和慢速移动环境下支持 384kb/s 的数据速率，室内静止环境下支持 2Mb/s 的高速数据传输，并保证可靠的服务质量。

参考答案

（70）B

试题（71）～（75）

Cloud computing is a phrase used to describe a variety of computing concepts that involve a large number of computers __（71）__ through a real-time communication network such as the Internet. In science, cloud computing is a __（72）__ for distributed computing over a network, and means the __(73)__ to run a program or application on many connected computers at the same time.

The architecture of a cloud is developed at three layers: infrastructure, platform, and application. The infrastructure layer is built with virtualized compute, storage, and network resources. The platform layer is for general-purpose and repeated usage of the collection of software resources. The application layer is formed with a collection of all needed software modules for SaaS applications. The infrastructure layer serves as the __（74）__ for building the platform layer of the cloud. In turn, the platform layer is a foundation for implementing the __（75）__ layer for SaaS applications.

（71）A．connected　　B．implemented
　　C．optimized　　D．virtualized

（72）A．replacement　　B．switch
　　C．substitute　　D．synonym(同义词)

（73）A．ability　　B．approach
　　C．function　　D．method

（74）A．network　　B．foundation
　　C．software　　D．hardware

（75）A．resource　　B．service
　　C．application　　D．software

参考译文

本题考查对英语资料的阅读理解。本段英文简要介绍云计算相关概念。

云计算是用来描述各种计算概念的短语，包括大量计算机通过网络相互连接以实现分布计算，意思是同时在很多互联的计算机上运行程序或应用的能力。

云的架构分为基础设施层、平台层和应用层三层。基础设施层由虚拟计算、存储和网络资源构成。平台层用于一组软件资源重复使用的通用目的。应用层由一组所需的软件模块构成，即软件即服务（SaaS）。基础设施层作为构建平台层的基础。相反，平台层是应用层的基础，为 SaaS 应用实现应用层。

参考答案

（71）A　　（72）D　　（73）A　　（74）B　　（75）C

第 22 章　2014 上半年软件设计师下午试题分析与解答

试题一（共 15 分）

阅读下列说明和图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

【说明】

某巴士维修连锁公司欲开发巴士维修系统，以维护与维修相关的信息。该系统的主要功能如下：

1）记录巴士 ID 和维修问题。巴士到车库进行维修，系统将巴士基本信息和 ID 记录在巴士列表文件中，将待维修机械问题记录在维修记录文件中，并生成维修订单。

2）确定所需部件。根据维修订单确定维修所需部件，并在部件清单中进行标记。

3）完成维修。机械师根据维修记录文件中的待维修机械问题，完成对巴士的维修，登记维修情况；将机械问题维修情况记录在维修记录文件中，将所用部件记录在部件清单中，并将所用部件清单发送给库存管理系统以对部件使用情况进行监控。巴士司机可查看已维修机械问题。

4）记录维修工时。将机械师提供的维修工时记录在人事档案中；将维修总结发送给主管进行绩效考核。

5）计算维修总成本。计算部件清单中实际所用部件、人事档案中所用维修工时的总成本；将维修工时和所用部件成本详细信息给会计进行计费。

现采用结构化方法对巴士维修系统进行分析与设计，获得如图 1-1 所示的上下文数据流图和图 1-2 所示的 0 层数据流图。

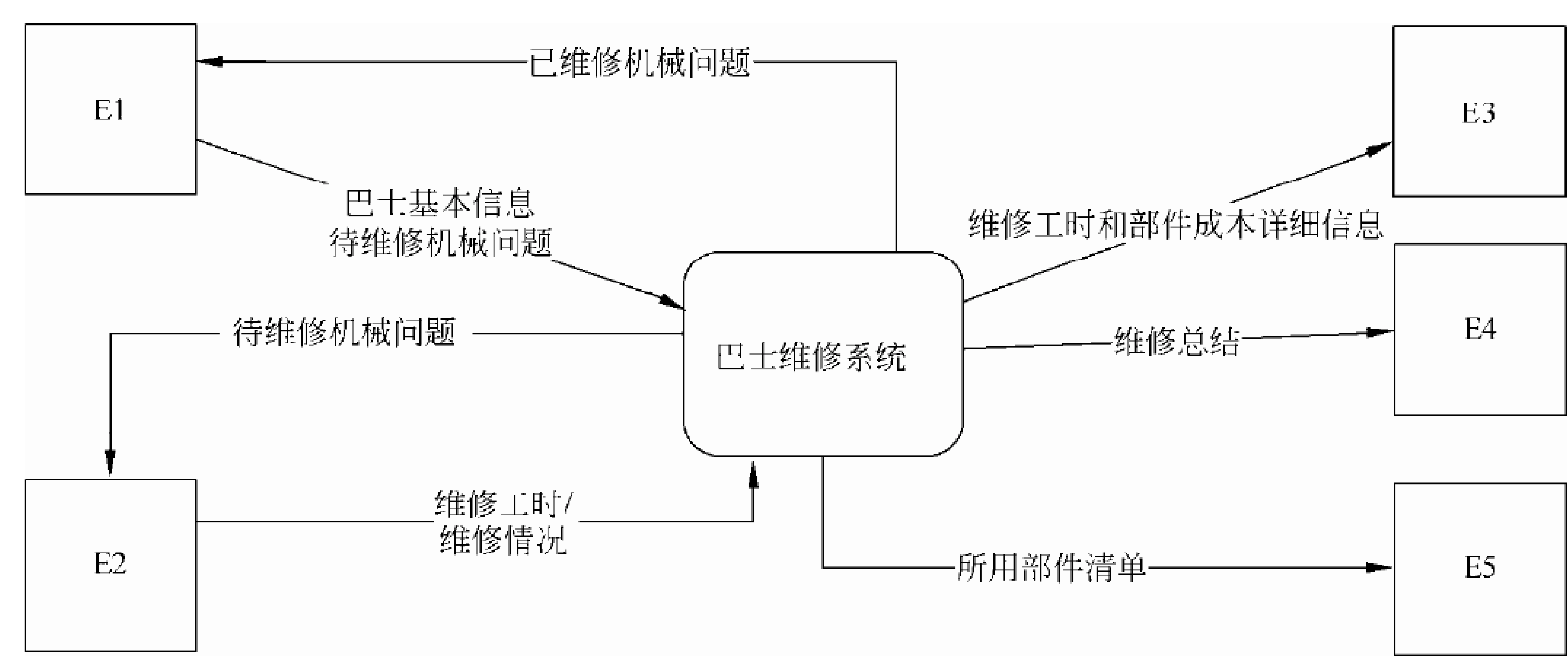


图 1-1　上下文数据流图

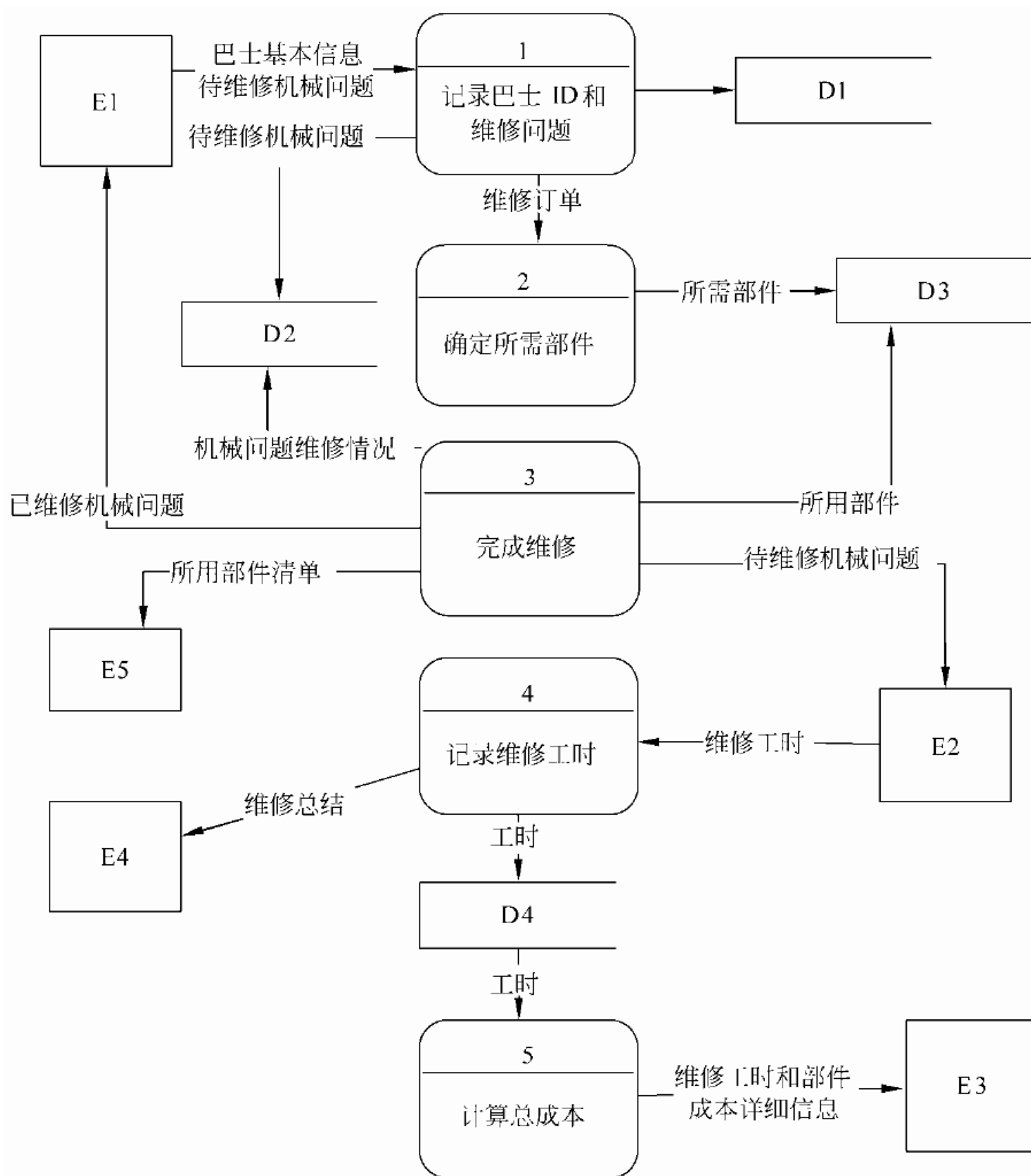


图 1-2　0 层数据流图

【问题 1】(5 分)

使用说明中的词语，给出图 1-1 中的实体 E1～E5 的名称。

【问题 2】(4 分)

使用说明中的词语，给出图 1-2 中的数据存储 D1～D4 的名称。

【问题 3】(3 分)

说明图 1-2 中所存在的问题。

【问题 4】(3 分)

根据说明和图中术语，采用补充数据流的方式，改正图 1-2 中的问题。要求给出所补充数据流的名称、起点和终点。

试题一分析

本题考查的是 DFD 的应用，属于比较传统的题目，考查点也与往年类似。

【问题 1】

本问题考查的是顶层 DFD。顶层 DFD 通常用来确定系统边界，其中只包含一个唯一的加工（即待开发的系统）、外部实体以及外部实体与系统之间的输入输出数据流。题目要求填充的正是外部实体。

从题干说明 1）没有明确说明由巴士到车库后由谁提供待维修问题，图 1-1 中的 E1，考察说明中 3）中最后一句说明“巴士司机可查看已维修机械问题”可以看出，从系统到巴士司机有输出数据流“已维修机械问题”，可知 E1 为巴士司机。从 2）中“机械师根据维修记录文件中的待维修机械问题，完成对巴士的维修，登记维修情况”；再看说明 4）中机械师提供维修工时，可以看出，从 E2 到系统有输入数据流“维修工时”、输出数据流“待维修机械问题”，可知 E2 为机械师，还将维修总结发送给主管，即系统到 E4 有输出数据流“维系总结”，可知 E4 为主管。从说明 5）将维修工时和所用部件成本详细信息给会计，从系统到 E3 有输出数据流“维修工时和所用部件成本详细信息”，可知 E3 为会计。说明 3）中将所用部件清单发送给库存管理系统以对部件使用情况进行监控，及系统到 E5 有输出数据流“所用部件清单”，可知 E5 为库存管理系统。

【问题 2】

本问题考查 0 层数据流图中的数据存储。系统中的主要功能与图 1-2 中的处理一一对应，1）对应处理“记录巴士 ID 和维修问题”，将巴士 ID 记录在巴士列表文件中，可知 D1 为巴士列表文件。说明 2）对应处理“确定所需部件”，将维修所需部件在部件清单中进行标记，可知以 D3 为部件清单。说明 1）中将待维修机械问题记录在维修记录文件中，可知 D2 为维修记录文件。说明 4）对应处理“记录维修工时”，描述了将机械师提供的维修工时记录在人事档案中，可以判定 D4 是人事档案。

【问题 3】

本问题考查 0 层数据流图中的数据流。分析图 1-2，可以发现，处理 3 只有输出数据流没有输入数据流，D2 和 D3 只有输入数据流，而没有输出流，造成黑洞。另外，对照图 1-2 和图 1-1，发现图 1-1 中从 E2 输入的数据流维修工时/维修情况，在图 1-2 中只有维修工时，造成父图与子图不平衡。

【问题 4】

针对【问题 3】分析图 1-2 中存在的问题，题目要求以补充数据流的方式解决，进一步分析说明，说明 3）对应处理“完成维修”，机械师根据维修记录文件中的待维修机械问题完成对巴士的维修，可知处理完成维修需要从维修记录文件读取待维修问题，补充一条从 D2 到处理 3 的数据流“待维修机械问题”。说明 5）对应处理“计算维修总成本”，需要计算部件清单中实际所用部件，补充从部件清单到计算总成本的数据流“实际

所用部件”。说明 3）中机械师要登记维修情况，判定图 1-2 中缺少了 E2 到处理 3 的数据流“维修情况”。

到此为止所有缺失的数据流都补齐了，也解决了【问题 3】中的平衡问题、处理只有输出数据流没有输入数据流的问题，D2 和 D3 也既有输入数据流，又有输出数据流。

试题一参考答案

【问题 1】

E1：巴士司机　　E2：机械师　　E3：会计

E4：主管　　E5：库存管理系统

【问题 2】

D1：巴士列表文件　　D2：维修记录文件

D3：部件清单　　D4：人事档案

【问题 3】

图 1-2 中处理 3 只有输出数据流，没有输入数据流。D2 和 D3 是黑洞，只有输入的数据流，没有输出的数据流。父图与子图不平衡，图 1-2 中没有图 1-1 中的数据流“维修情况”。

【问题 4】

数据流名称	起　　点	终　　点
待维修机械问题	D2 或维修记录文件	3 或完成维修
实际所用部件	D3 或部件清单	5 或计算总成本
维修情况	E2 或机械师	3 或完成维修

试题二（共 15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某家电销售电子商务公司拟开发一套信息管理系统，以方便对公司的员工、家电销售、家电厂商和客户等进行管理。

【需求分析】

（1）系统需要维护电子商务公司的员工信息、客户信息、家电信息和家电厂商信息等。员工信息主要包括：工号、姓名、性别、岗位、身份证号、电话、住址，其中岗位包括部门经理和客服等。客户信息主要包括：客户 ID、姓名、身份证号、电话、住址、账户余额。家电信息主要包括：家电条码、家电名称、价格、出厂日期、所属厂商。家电厂商信息包括：厂商 ID、厂商名称、电话、法人代表信息、厂址。

（2）电子商务公司根据销售情况，由部门经理向家电厂商订购各类家电。每个家电厂商只能由一名部门经理负责。

（3）客户通过浏览电子商务公司网站查询家电信息，与客服沟通获得优惠后，在线购买。

【概念模型设计】

根据需求阶段收集的信息，设计的实体联系图（不完整）如图 2-1 所示。

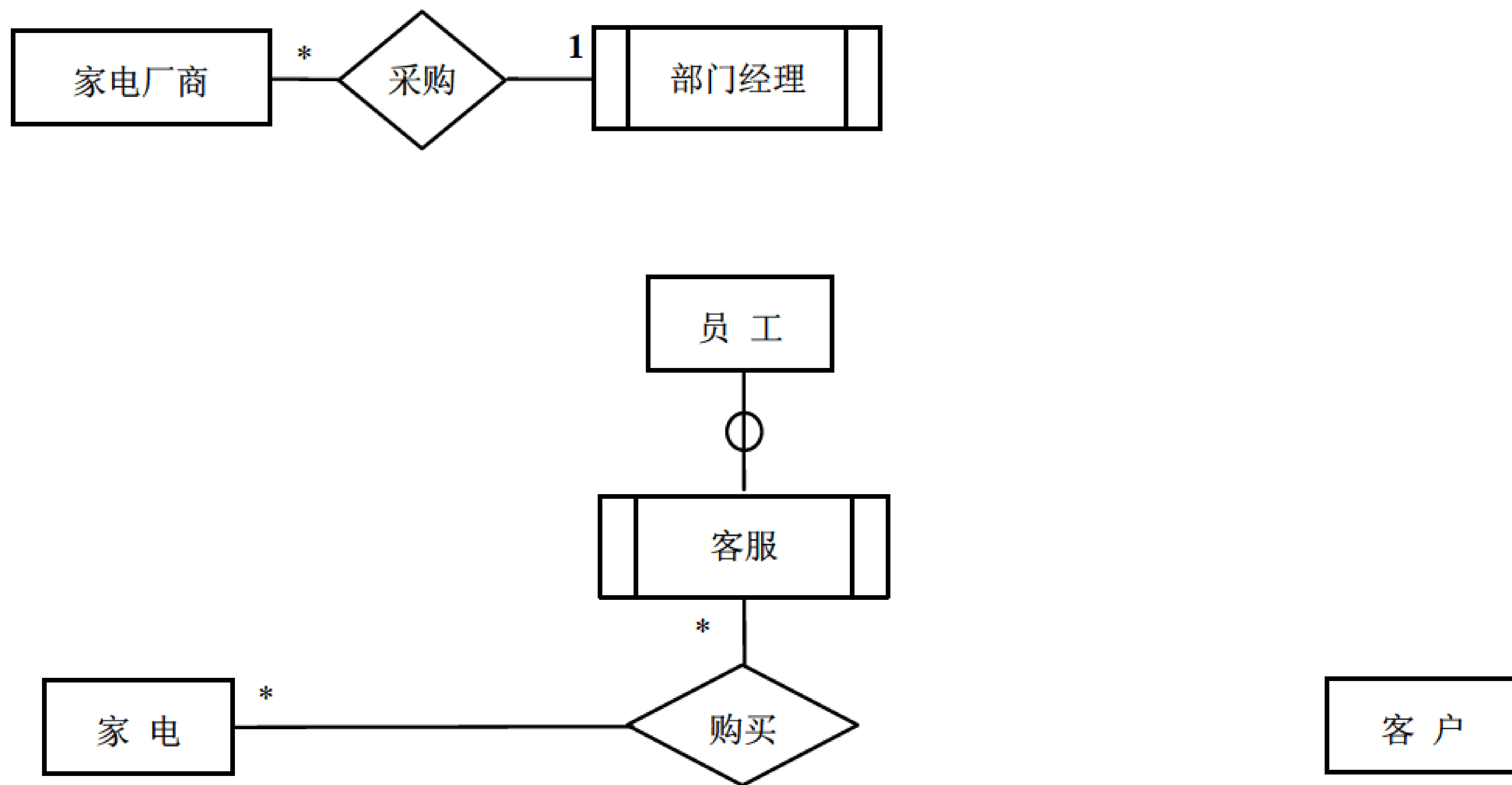


图 2-1 实体联系图

【逻辑结构设计】

根据概念模型设计阶段完成的实体联系图，得出如下关系模式（不完整）：

客户（<u>客户 ID</u>、姓名、身份证号、电话、住址、账户余额）

员工（<u>工号</u>、姓名、性别、岗位、身份证号、电话、住址）

家电（家电条码、家电名称、价格、出厂日期、____（1）____）

家电厂商（厂商 ID、厂商名称、电话、法人代表信息、厂址、____（2）____）

购买（订购单号、________（3）________、金额）

【问题1】（6分）

补充图 2-1 中的联系和联系的类型。

【问题2】（6分）

根据图 2-1，将逻辑结构设计阶段生成的关系模式中的空（1）～（3）补充完整。用下划线指出“家电”“家电厂商”和“购买”关系模式的主键。

【问题3】（3分）

电子商务公司的主营业务是销售各类家电，对账户有余额的客户，还可以联合第三方基金公司提供理财服务，为此设立客户经理岗位。客户通过电子商务公司的客户经理和基金公司的基金经理进行理财。每名客户只由一名客户经理和一名基金经理负责，客

户经理和基金经理均可负责多名客户。请根据该要求，对图 2-1 进行修改，画出修改后的实体间联系和联系的类型。

试题二分析

本题考查数据库设计，属于比较传统的题目，考查点也与往年类似。

【问题 1】

本问题考查数据库的概念结构设计，题目要求补充完整实体联系图中的联系和联系的类型。

根据题目的需求描述可知，一个家电厂商可以供应多台家电，而一台家电只能对应一个家电厂商，因此“家电厂商”和“家电”之间存在“供应”联系，联系的类型为一对多（1:*，或 1:m）。

根据题目的需求描述可知，“员工”和“部门经理”之间存在一个包含关系。

根据题目的需求描述可知，“客户”“客服”和“家电”之间存在“购买”联系，联系的类型为多对多对多（*:*:*，或 m:n:o）。

【问题 2】

本问题考查数据库的逻辑结构设计，题目要求补充完整各关系模式，并给出各关系模式的主键。

根据实体联系图和需求描述，“家电”和“家电厂商”存在多对一的关系，在家电关系中需要记录家电厂商的主键，也就是“厂商 ID”。所以，对于“家电”关系模式，需补充属性“厂商 ID”。“家电条码”为“家电”关系的主键。

根据实体联系图和需求描述，“家电厂商”和“部门经理”之间存在多对一的关系，在家电厂商关系中需要记录部门经理的主键，也就是“部门经理工号”（或“经理工号”、或“员工工号”）。“厂商 ID”为“家电厂商”的主键。

根据实体联系图和需求描述，“客户”“客服”和“家电”之间的多对多对多的“购买”联系。因为是多对多对多联系，所以“购买”联系需要单独作为一个关系，这个关系需要记录“客户”“客服”和“家电”的主键。所以，对于“购买”关系模式，需补充属性“客户 ID”“客服工号”和“家电条码”。“订购单号”为“购买”的主键。

【问题 3】

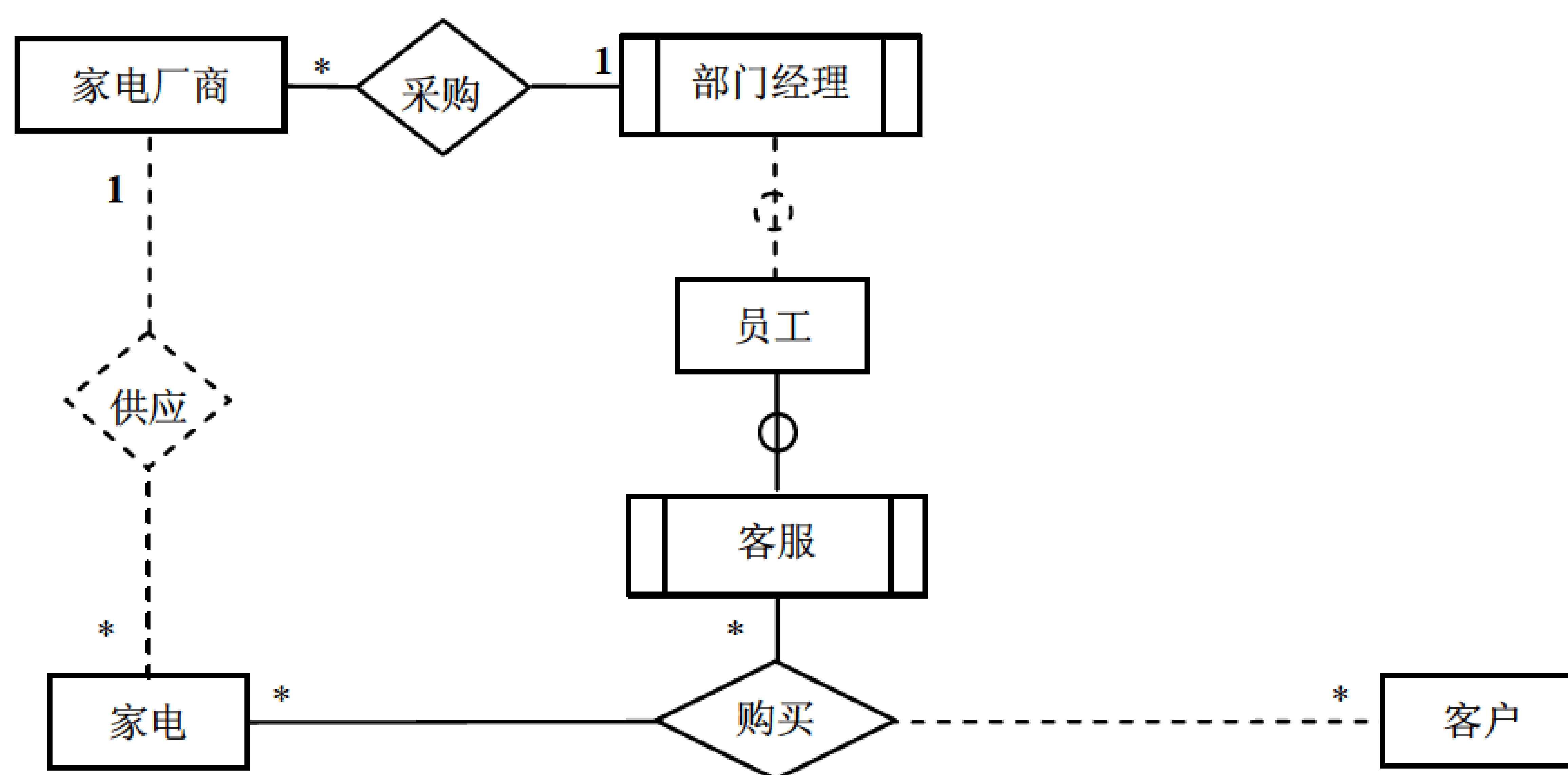
本问题考查数据库的概念结构设计，根据新增的需求增加实体联系图中的实体的联系和联系的类型。

根据问题描述，需要新增“客户经理”，包含于“员工”。

根据问题描述，客户只由一名客户经理和一名基金经理负责，客户经理和基金经理均可负责多名客户，所以“客户”、“客户经理”和“基金经理”之间存在一个“理财”联系，联系的类型为多对 1 对 1（*:1:1，或 m:1:1）。

试题二参考答案

【问题 1】



【问题 2】

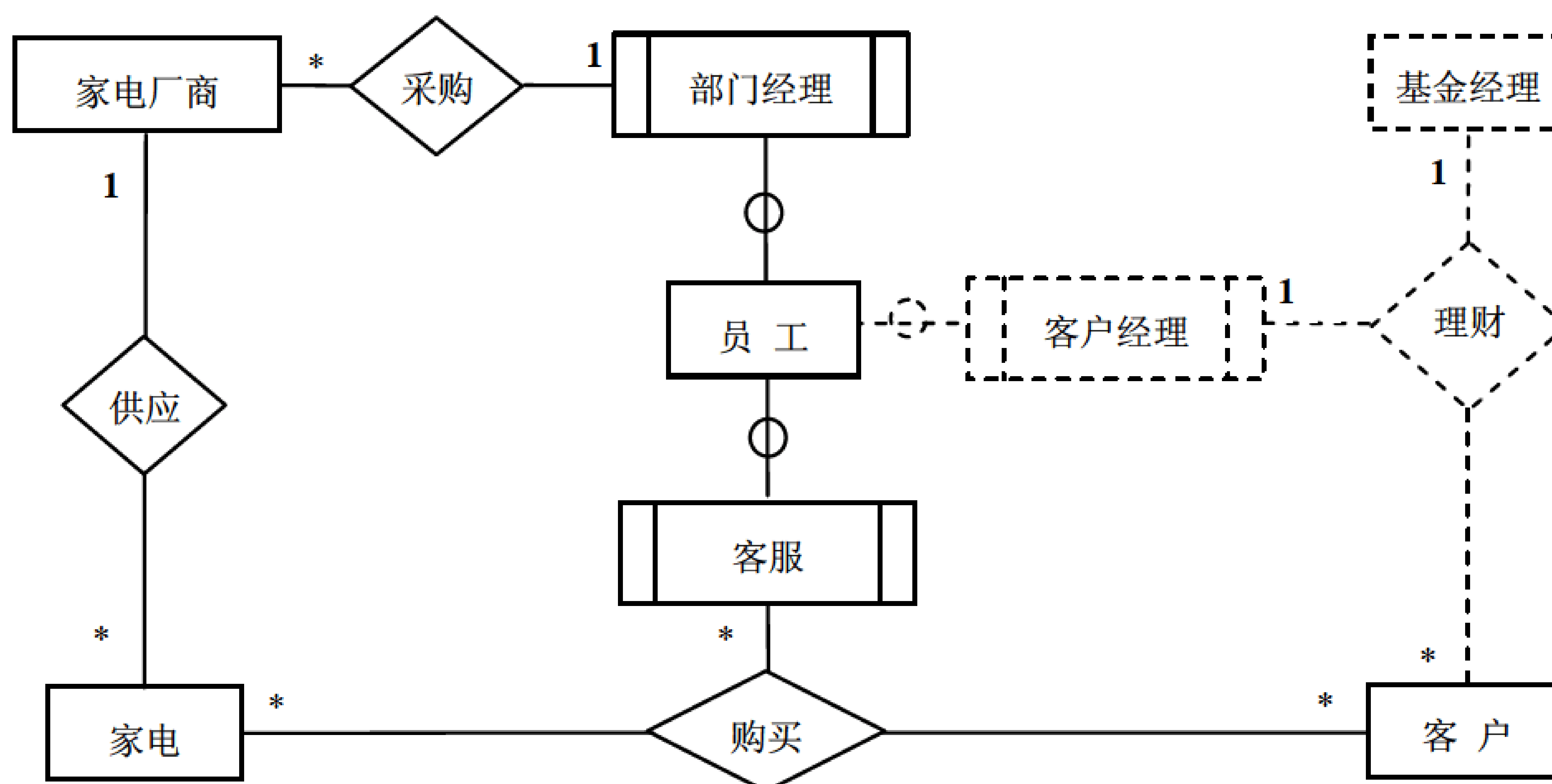
（1）厂商 ID

（2）部门经理工号 或 经理工号 或 员工工号

（3）客户 ID、客服工号、家电条码

关系模式	主键
家电	家电条码
家电厂商	厂商 ID
购买	订购单号

【问题 3】



试题三（共 15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某高校图书馆欲建设一个图书馆管理系统，目前已经完成了需求分析阶段的工作。功能需求均使用用例进行描述，其中用例“借书（Check Out Books）”的详细描述如下。

参与者：读者（Patron）。

典型事件流：

1．输入读者 ID；

2．确认该读者能够借阅图书，并记录读者 ID；

3．输入所要借阅的图书 ID；

4．根据图书目录中的图书 ID 确认该书可以借阅，计算归还时间，生成借阅记录；

5．通知读者图书归还时间。

重复步骤 3~5，直到读者结束借阅图书。

备选事件流：

2a．若读者不能借阅图书，说明读者违反了图书馆的借书制度（例如，没有支付借书费用等）

① 知读者不能借阅，并说明拒绝借阅的原因；

② 用例结束。

4a．读者要借阅的书无法外借

① 知读者本书无法借阅；

② 到步骤 3。

说明：图书的归还时间与读者的身份有关。如果读者是教师，图书可以借阅一年；如果是学生，则只能借阅 3 个月。读者 ID 中包含读者身份信息。

现采用面向对象方法开发该系统，得到如图 3-1 所示的系统类模型（部分）；以及如图 3-2 所示的系统操作“checkOut(bookID)（借书）”的通信图（或协作图）。

【问题 1】（8 分）

根据说明中的描述，以及图 3-1 和图 3-2，给出图 3-1 中 C1～C4 处所对应的类名（类名使用图 3-1 和图 3-2 中给出的英文词汇）。

【问题 2】（4 分）

根据说明中的描述，以及图 3-1 和图 3-2，给出图 3-2 中 M1～M4 处所对应的方法名（方法名使用图 3-1 和图 3-2 中给出的英文词汇）。

【问题 3】（3 分）

用例“借书”的备选事件流 4a 中，根据借书制度来判定读者能否借阅图书。若图书馆的借书制度会不断地扩充，并需要根据图书馆的实际运行情况来调整具体使用哪些制度。为满足这一要求，在原有类设计的基础上，可以采用何种设计模式？简要说明原因。

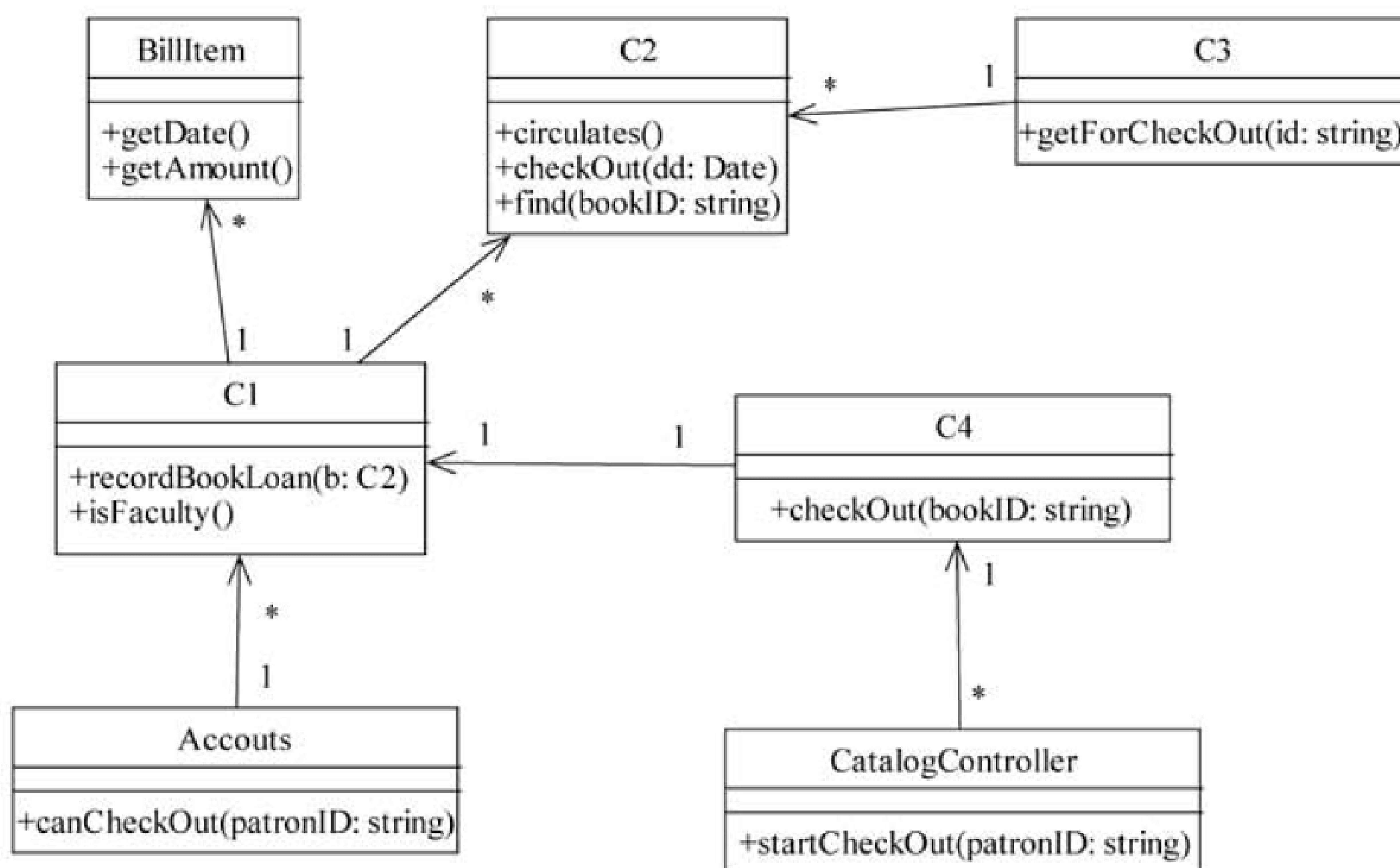


图 3-1　系统类模型（部分）

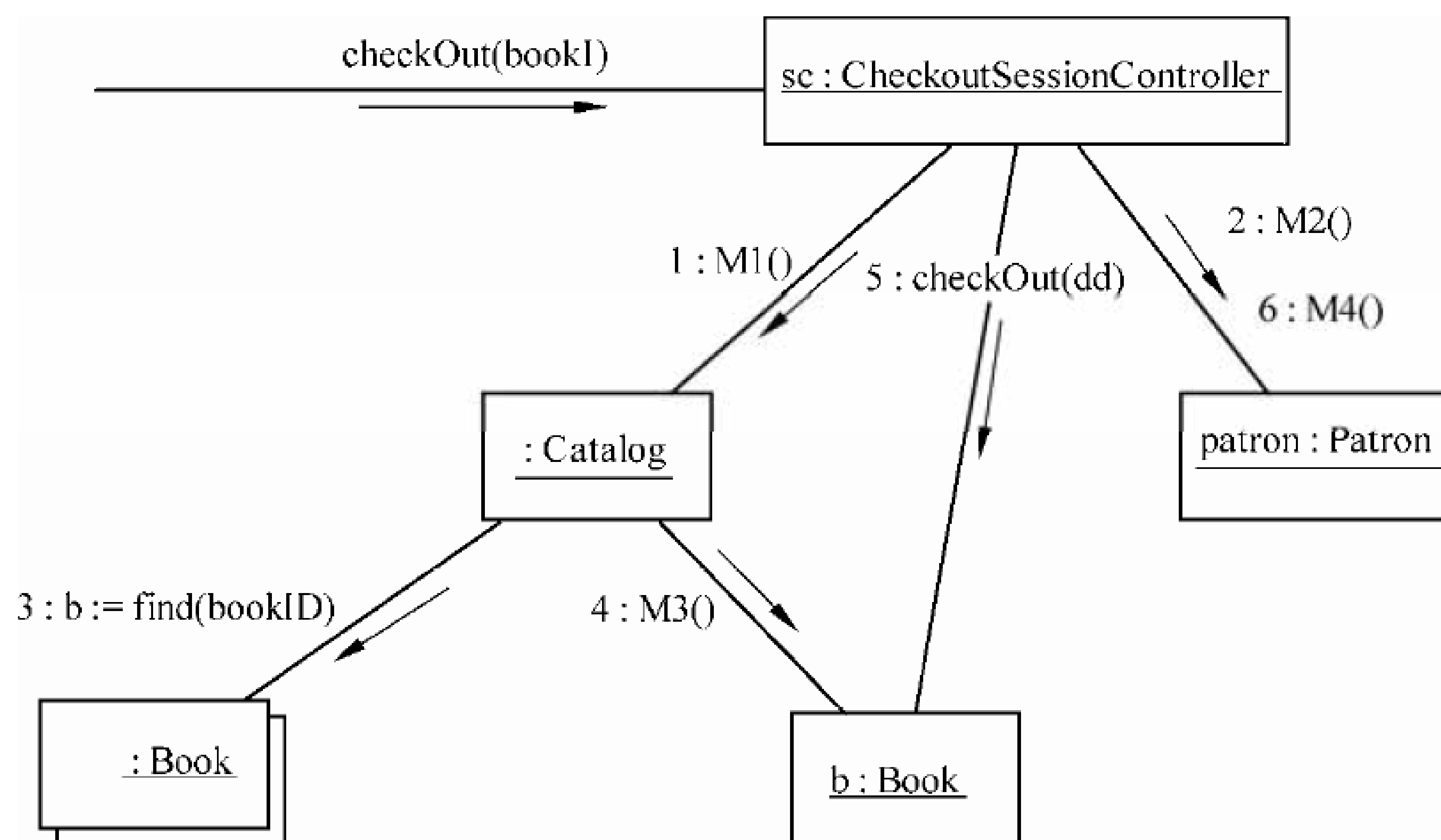


图 3-2　统操作 checkOut 的通信图

试题三分析

本题属于经典的考题，主要考查面向对象分析方法以及 UML 类图和通信图的相关知识。

【问题 1】

说明中给出了一个具体用例的详细描述，给出了其中的一个系统操作“checkout bookID)（借书）”的通信图，需要考生利用通信图中的信息来补充类图中缺失的部分。

通信图（communication diagram）强调收发消息的对象的结构组织，在早期的版本

中也被称作协作图。通信图强调参加交互的对象的组织。产生一张通信图，首先要将参加交互的对象作为图的顶点，然后把连接这些对象的链表示为图的弧，最后用对象发送和接收的消息来修饰这些链。这就提供了在协作对象的结构组织的语境中观察控制流的一个清晰的可视化轨迹。

消息 checkOut(bookID)的接收者是类 CheckoutSessionController 的对象，说明类 CheckoutSessionController 中应该包含这一方法，否则无法响应该条消息。由图 3-1 可知，C4 处所代表的类应该是 CheckoutSessionController。

消息 find(bookID)的接收者是类 Book，同理，由图 3-1 可知，C2 处对应的类应该是 Book。

根据用例描述，图书信息是包含在图书目录中，所以 C3 处对应的类应该是 Catalog，C1 处对应的就应该是 Patron 了。

【问题 2】

图 3-1 填充完整之后，图 3-2 的空缺就比较容易填写了。在通信图中，对象之间传递的消息就对应着接收对象中的方法。M1 对应的就是类 Catalog 中的方法，由图 3-1 可知，M1 对应的是 getForCheckOut。

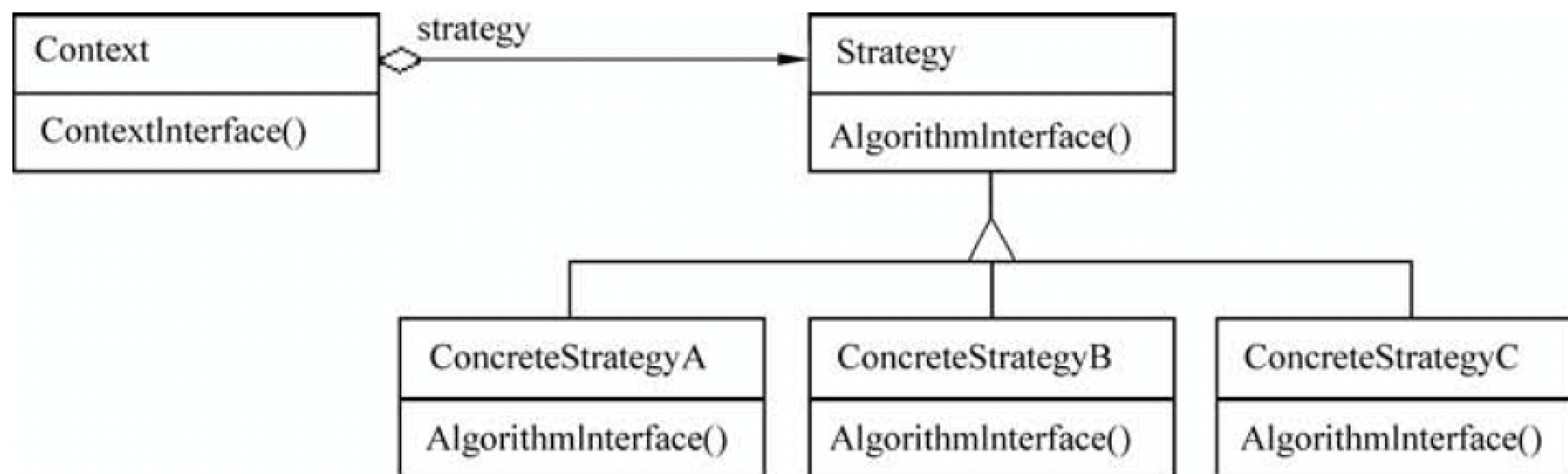
M3 对应的应该是类 Book 中的方法。由图 3-1 可知，Book 中有 3 个方法，find 和 checkOut 已经出现在通信图上了，所以 M3 应该是 circulates。

M2 和 M4 是类 Patron 中的方法。Patron 中有 2 个方法。通信图中的消息是有序号的，这个序号表示了消息的时间顺序，也就是说发送 M2 的时间要早于消息 M4，因此必须区分类 Patron 中两个方法使用的先后顺序。在用例描述中特别指出：图书的归还时间与读者的身份有关。计算还书及借书费用时，需先确定读者的身份，因此方法 isFaculty 应该先被调用，所以 M2 对应 isFaculty，M4 对应 recordBookLoan。

【问题 3】

本题在设计类时使用到了策略模式。

策略模式定义了一系列的算法，把它们一个个封装起来，并且使它们可以相互替换。此模式使得算法可以独立于使用它们的客户而变化。策略模式的结构如下图所示。



其中：

- Strategy（策略）定义所有支持的算法的公共接口。Context 使用这个接口来调用某 ConcreteStrategy 定义的算法。
- ConcreteStrategy（具体策略）以 Strategy 接口实现某具体算法。
- Context（上下文）用一个 ConcreteStrategy 对象来配置；维护一个对 Strategy 对象的引用；可定义一个接口来让 Strategy 访问它的数据。

Strategy 模式适用于：

- 许多相关的类仅仅是行为有异。“策略”提供了一种用多个行为中的一个行为来配置一个类的方法。
- 需要使用一个算法的不同变体。例如，定义一些反应不同空间的空间/时间权衡的算法。当这些变体实现为一个算法的类层次时，可以使用策略模式。
- 算法使用客户不应该知道的数据。可使用策略模式以避免暴露复杂的、与算法相关的数据结构。
- 一个类定义了多种行为，并且这些行为在这个类的操作中以多个条件语句的形式出现，将相关的条件分支移入它们各自的 Strategy 类中，以代替这些条件语句。

试题三参考答案

【问题 1】

C1：Patron　　C2：Book

C3：Catalog　　C4：CheckoutSessionController

【问题 2】

M1：getForCheckOut　　M2：isFaculty

M3：circulates　　M4：recordBookLoan

【问题 3】

解答：策略模式。

策略模式定义了一系列算法，并将每个算法封装起来，而且使它们可以相互替换。策略模式让算法独立于使用它们的客户而变化。适用于需要在不同情况下使用不同的策略（算法），或者策略还可能在未来用其他方式来实现。

试题四（共 15 分）

阅读下列说明和 C 代码，回答问题 1 至问题 3，将解答写在答题纸的对应栏内。

【说明】

采用归并排序对 n 个元素进行递增排序时，首先将 n 个元素的数组分成各含 n/2 个元素的两个子数组，然后用归并排序对两个子数组进行递归排序，最后合并两个已经排好序的子数组得到排序结果。

下面的 C 代码是对上述归并算法的实现，其中的常量和变量说明如下：

arr：待排序数组

p, q, r：一个子数组的位置为从 p 到 q，另一个子数组的位置为从 q+1 到 r

begin, end：待排序数组的起止位置

left, right：临时存放待合并的两个子数组

n1, n2：两个子数组的长度

i, j, k：循环变量

mid：临时变量

【C 代码】

```
#include<stdio.h>
#include<stdlib.h>
#define MAX 65536
void merge(int arr[],int p,int q,int r){
    int *left, *right;
    int n1, n2, i, j, k;
    n1 = q - p + 1;
    n2 = r - q;
    if((left = (int*)malloc((n1 + 1)*sizeof(int))) == NULL){
        perror("malloc error");
        exit(1);
    }
    if((right = (int*)malloc((n2 + 1)*sizeof(int))) == NULL){
        perror("malloc error");
        exit(1);
    }
    for(i = 0; i < n1; i++){
        left[i] = arr[p + i];
    }
    left[i] = MAX;
    for(i = 0; i < n2; i++){
        right[i] = arr[q + i + 1];
    }
    right[i] = MAX;
    i = 0; j = 0;
    for(k = p; ___(1)___; k++) {
        if(left[i] > right[j]) {
            _____(2)_____;
            j++;
        }else{
            arr[k] = left[i];
            i++;
```

```
            }
        }
    }
    void mergeSort(int arr[],int begin,int end){
        int mid;
        if( ___(3)___ ) {
            mid = (begin + end) / 2;
            mergeSort(arr,begin,mid);
            ____(4)____;
            merge(arr,begin,mid,end);
        }
    }
```

【问题 1】(8 分)

根据以上说明和 C 代码，填充 C 代码中的空（1）～（4）。

【问题 2】(5 分)

根据题干说明和以上 C 代码，算法采用了___(5)___算法设计策略。

分析时间复杂度时，列出其递归式为___(6)___，解得渐进时间复杂度为___(7)___（用 O 符号表示）。空间复杂度为___(8)___（用 O 符号表示）。

【问题 3】(2 分)

两个长度分别为 n1 和 n2 的已排好序的子数组进行归并，根据上述 C 代码，则元素之间比较次数为___(9)___。

试题四分析

本题考查算法设计、分析和 C 程序实现的知识，属于传统题目，考查点也与往年类似。

归并排序是一种经典的排序算法，基本思想：把 n 个元素构成的数组分成两个 n/2 个元素构成的子数组，再进一步划分，一直到每个子数组仅包含 1 个元素，此时再把两两有序的数组合并成更大的有序数组，一直到整个数组有序为止。

【问题 1】

本问题考查算法的实现。C 程序中有两个函数，merge 函数将两个有序数组合并成一个更大的有序数组。归并过程是首先将两个有序的子数组的元素分别放到 left 和 right 数组中，然后依次比较这两个数组中的元素，从小到大把元素放到 arr 数组的特定元素中。包含空格（1）的 for 循环中，给出了将 left 和 right 元素放入 arr 中，放入的位置是从 p 到 r，因此，空格（1）填写 k <= r。在比较 left[i] 和 right[j]元素时，若 left[i] > right[j]，则应该把 right[j]的值放入 arr[k]中，因此空格（2）填写 arr[k] = right[j]。mergeSort 函数进行数组的排序。若数组元素个数大于 1，则继续划分，因此空格（3）填写 begin < end。将数组从 arr[begin]到 arr[end]划分为 arr[begin]到 arr[mid]和 arr[mid + 1]到 arr[end]两个部分，因此空格（4）填写 mergeSort(arr,mid + 1,end)。

【问题 2】

本问题考查算法的设计策略和时间复杂度，归并排序算法是一个典型的分治算法。每次将一个规模为 n 的问题变成两个规模为 n/2 的子问题，划分时直接从中间分开，因此划分采用 O(1)的时间，然后是递归求解两个子问题，根据 merge 函数代码，合并的时间是线性时间 O(n)。因此递归式为 $T(n) = 2T(n / 2) + f(n)$，f(n)为线性函数。用主方法求解，得到时间复杂度为 O(nlgn)。由于在归并过程中，需要 left 和 right 两个辅助数组，其规模为待排序的数组长度，即 O(n)。

【问题 3】

本问题考查对算法的进一步分析。元素之间的比较次数就是 merge 函数的最后一个 for 循环体执行的次数，由于 k 从 p 到 r，故循环体执行次数为 $r-p+1$ 次，即 $n1 + n2$ 次。

试题四参考答案

【问题 1】

（1）k <= r　或　k < r+1　　（2）arr[k] = right[j]

（3）begin < end　　（4）mergeSort(arr,mid + 1,end)

【问题 2】

（5）分治

（6）T(n)=2T(n/2)+ n　或　T(n)=2T(n/2)+f(n)（f(n)为线性函数）

（7）O(nlgn)

（8）O(n)

【问题 3】

（9）n1+n2

试题五（共 15 分）

阅读下列说明和 C++代码，将应填入<u>（n）</u>处的字句写在答题纸的对应栏内。

【说明】

某实验室欲建立一个实验室环境监测系统，能够显示实验室的温度、湿度以及洁净度等环境数据。当获取到最新的环境测量数据时，显示的环境数据能够更新。

现在采用观察者（Observer）模式来开发该系统。观察者模式的类图如图 5-1 所示。

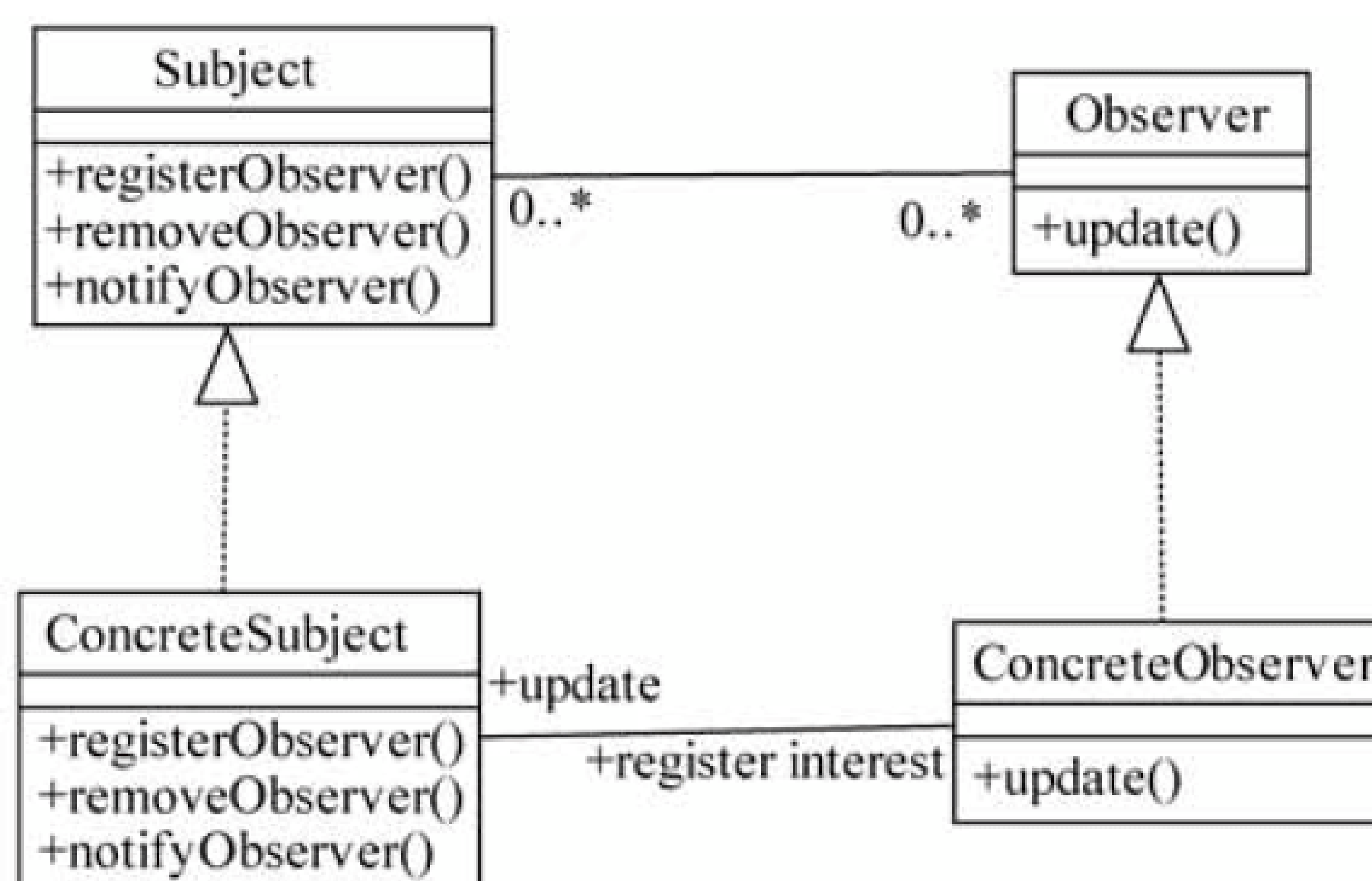


图 5-1　观察者模式类图

【C++代码】

```
#include <iostream>
#include <vector>
using namespace std;
class Observer {
public:
  virtual void update(float temp, float humidity, float cleanness) = 0;
};
class Subject {
public:
  virtual void registerObserver(Observer* o) = 0;//注册对主题感兴趣的观察者
  virtual void removeObserver(Observer* o) = 0; //删除观察者
  virtual void notifyObservers() = 0;           //当主题发生变化时通知观察者
};
class EnvironmentData : public ___(1)___ {
private:
  vector<Observer*> observers;
  float temperature, humidity, cleanness;
public:
  void registerObserver(Observer* o) {   observers.push_back(o);  }
  void removeObserver(Observer* o) {  /* 代码省略 */  }
  void notifyObservers() {
for(vector<Observer*>::const_iterator it = observers.begin(); it !=
observers.end(); it++)
{   ___(2)___;   }
}
  void measurementsChanged() {  ___(3)___;  }
  void setMeasurements(float temperature, float humidity, float cleanness) {
    this->temperature = temperature;
    this->humidity = humidity;
    this->cleanness = cleanness;
    ___(4)___;
  }
};
class CurrentConditionsDisplay : public ___(5)___{
private:
    float temperature, humidity, cleanness;
    Subject* envData;
public:
    CurrentConditionsDisplay(Subject* envData) {
      this->envData = envData;
```

```
        ____(6)____;
    }
    void update(float temperature, float humidity, float cleanness) {
        this->temperature = temperature;
        this->humidity = humidity;
        this->cleanness = cleanness;
        display();
    }
    void display() {   /* 代码省略 */  }
};
int main() {
    EnvironmentData* envData = new EnvironmentData();
    CurrentConditionsDisplay* currentDisplay = new CurrentConditions
    Display(envData);
    envData->setMeasurements(80, 65, 30.4f);
    return 0;
}
```

试题五分析

本题考察观察者（Observer）模式的概念及应用。

观察者模式定义对象间的一种一对多的依赖关系，当一个对象的状态发生改变时，所有依赖于它的对象都得到通知并被自动更新。

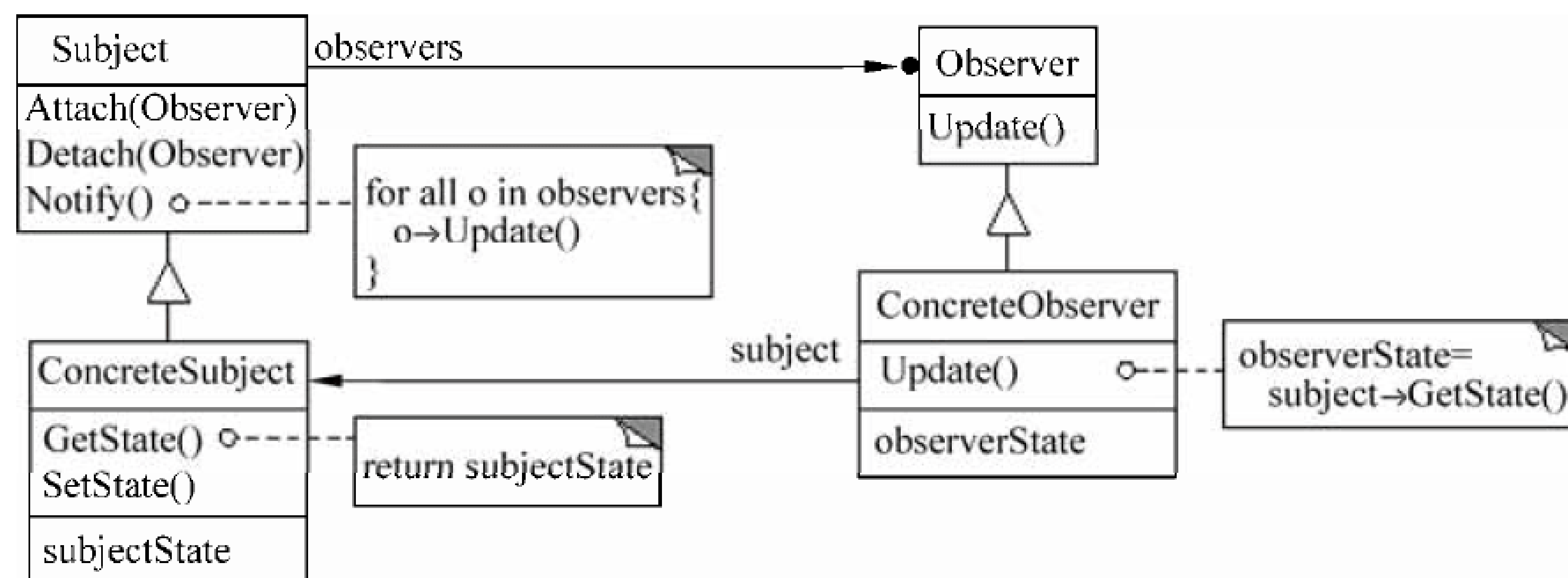
Observer 模式适用于：

① 当一个抽象模型有两个方面，其中一个方面依赖于另一个方面。将这两者封装在独立地对象中以使它们可以各自独立地改变和复用。

② 当对一个对象的改变需要同时改变其他对象，而不知道具体有多少对象有待改变时。

③ 当一个对象必须通知其他对象，而它又不能假定其他对象是谁，即：不希望这些对象是紧耦合的。

观察者模式的结构如下图所示，其中：



Subject（主题）知道它的观察者，可以有任意多个观察者观察同一个目标；提供注册和删除观察者对象的接口。

Observer（观察者）为那些在目标发生改变时需获得通知的对象定义一个更新接口。

ConcreteSubject（具体主题）将有关状态存入一个 ConcreteObserver 对象；当它的状态发生改变时，向它的各个观察者发出通知。

ConcreteObserver（具体观察者）维护一个指向 ConcreteSubject 对象的引用；存储有关状态，这些状态应与主题的状态保持一致；实现 Observer 的更新接口以使自身状态与主题的状态保持一致。

在本题的说明中，给出了观察者模式的结构图，答题时需要首先确定程序中的类与观察者模式结构的对应关系，也就是要找到哪些是 ConcreteSubject，哪些是 ConcreteObserver。由程序上下文可以判断出，类 EnvironmentData 对应的是 ConcreteSubject，类 CurrentConditionsDisplay 对应的是 ConcreteObserver。根据类图，它们分别为 Subject 和 Observer 的子类。由此可以，空（1）和空（5）应分别填写 Subject 和 Observer。

空（2）要求给出方法 notifyObservers 的实现，其功能是在主题发生变化时通知观察者。通知的实现是通过向对该主题感兴趣的所有观察者发送 update 消息实现的。对主题 EnvironmentData 感兴趣的观察者由向量 observers 表示，所以在 notifyObservers 方法中，就是对向量 observers 中的每个成员发送 update 消息，因此空（2）应填写 (*it)->update(temperature, humidity, cleanness)。

方法 measurementsChanged 表示主题发生了变化，这时应该通知对应的观察者，所以空（3）处应填写 notifyObservers()。

方法 setMeasurements 用于设置发生变化后的主题内容，所以空（4）处应填写 measurementsChanged()。

CurrentConditionsDisplay 是对主题 EnvironmentData 感兴趣的一个观察者，要能够获得主题的变化，需要首先将自己注册为该主题的观察者，这个注册行为在其构造函数中完成。因此空（6）处应填写 envData->registerObserver(this)。

试题五参考答案

（1）Subject

（2）(*it)->update(temperature, humidity, cleanness)

（3）notifyObservers()

（4）measurementsChanged()

（5）Observer

（6）envData->registerObserver(this)

试题六（共 15 分）

阅读下列说明和 Java 代码，将应填入<u>（n）</u>处的字句写在答题纸的对应栏内。

【说明】

某实验室欲建立一个实验室环境监测系统，能够显示实验室的温度、湿度以及洁净度等环境数据。当获取到最新的环境测量数据时，显示的环境数据能够更新。

现在采用观察者（Observer）模式来开发该系统。观察者模式的类图如图 6-1 所示。

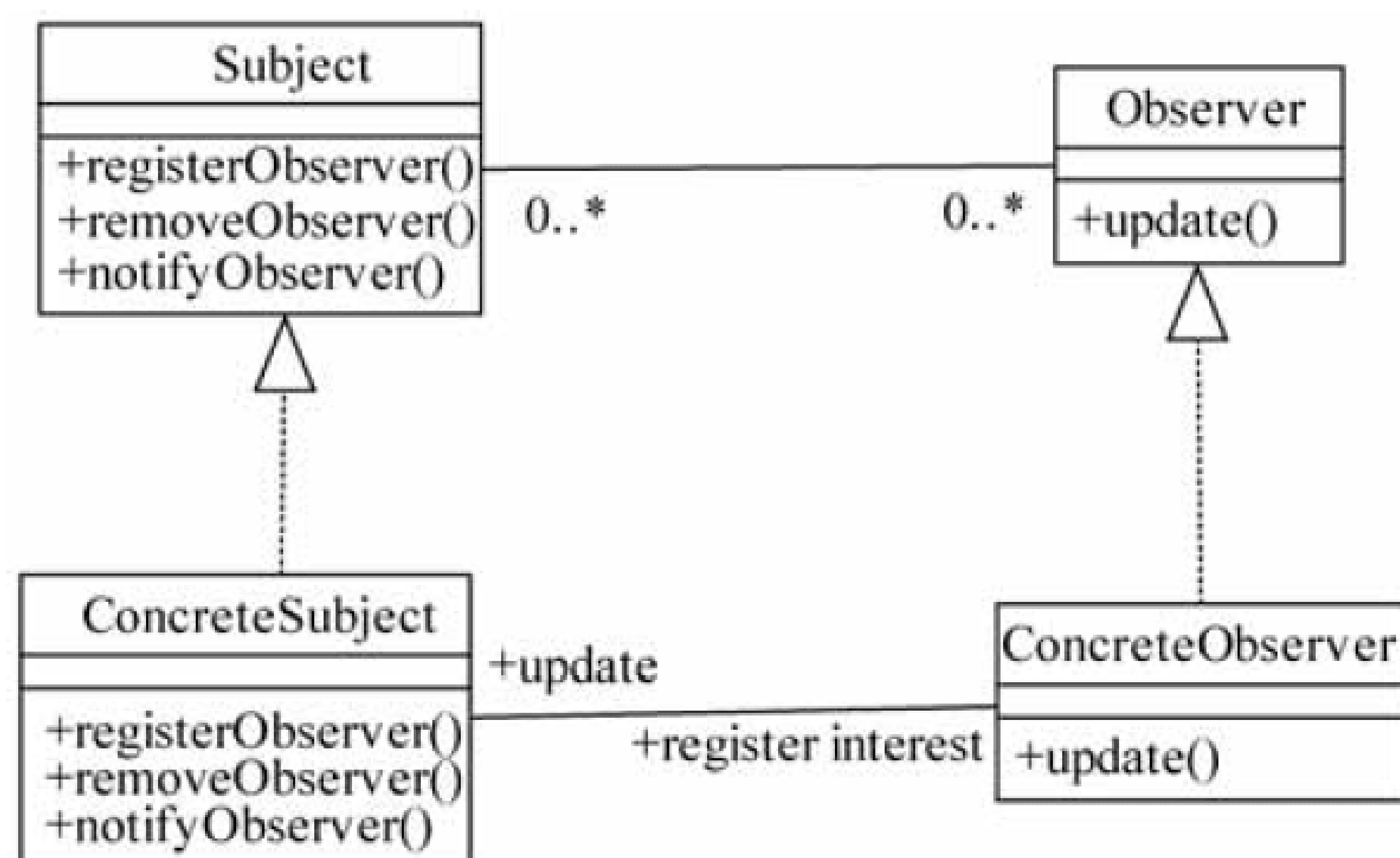


图 6-1 观察者模式类图

【Java 代码】

```
import java.util.*;
interface Observer {
  public void update(float temp, float humidity, float cleanness);
}
interface Subject {
  public void registerObserver(Observer o); //注册对主题感兴趣的观察者
  public void removeObserver(Observer o);   //删除观察者
  public void notifyObservers();            //当主题发生变化时通知观察者
}
class EnvironmentData implements ___(1)___{
  private ArrayList observers;
  private float temperature, humidity, cleanness;
  public EnvironmentData() {   observers = new ArrayList();  }
  public void registerObserver(Observer o) {  observers.add(o);  }
  public void removeObserver(Observer o) {  /* 代码省略 */  }
  public void notifyObservers() {
    for (int i = 0; i < observers.size(); i++) {
      Observer observer = (Observer)observers.get(i);
      ___(2)___;
    }
  }
```

```
  public void measurementsChanged() { ____(3)____; }
public void setMeasurements(float temperature, float humidity, float
cleanness) {
    this.temperature = temperature;
    this.humidity = humidity;
    this.cleanness = cleanness;
    ____(4)____;
  }
}
class CurrentConditionsDisplay implements ____(5)____{
    private float temperature;
    private float humidity;
    private float cleanness;
    private Subject envData;
    public CurrentConditionsDisplay(Subject envData) {
      this.envData = envData;
      ____(6)____;
}
    public void update(float temperature, float humidity, float cleanness) {
      this.temperature = temperature;
      this.humidity = humidity;
      this.cleanness = cleanness;
      display();
}
    public void display() { /* 代码省略 */ }
}
class EnvironmentMonitor{
  public static void main(String[] args) {
    EnvironmentData envData = new EnvironmentData();
    CurrentConditionsDisplay currentDisplay = new CurrentConditions
    Display(envData);
    envData.setMeasurements(80, 65, 30.4f);
  }
}
```

试题六分析

本题考查观察者（Observer）模式的概念及应用。

观察者模式定义对象间的一种一对多的依赖关系，当一个对象的状态发生改变时，所有依赖于它的对象都得到通知并被自动更新。

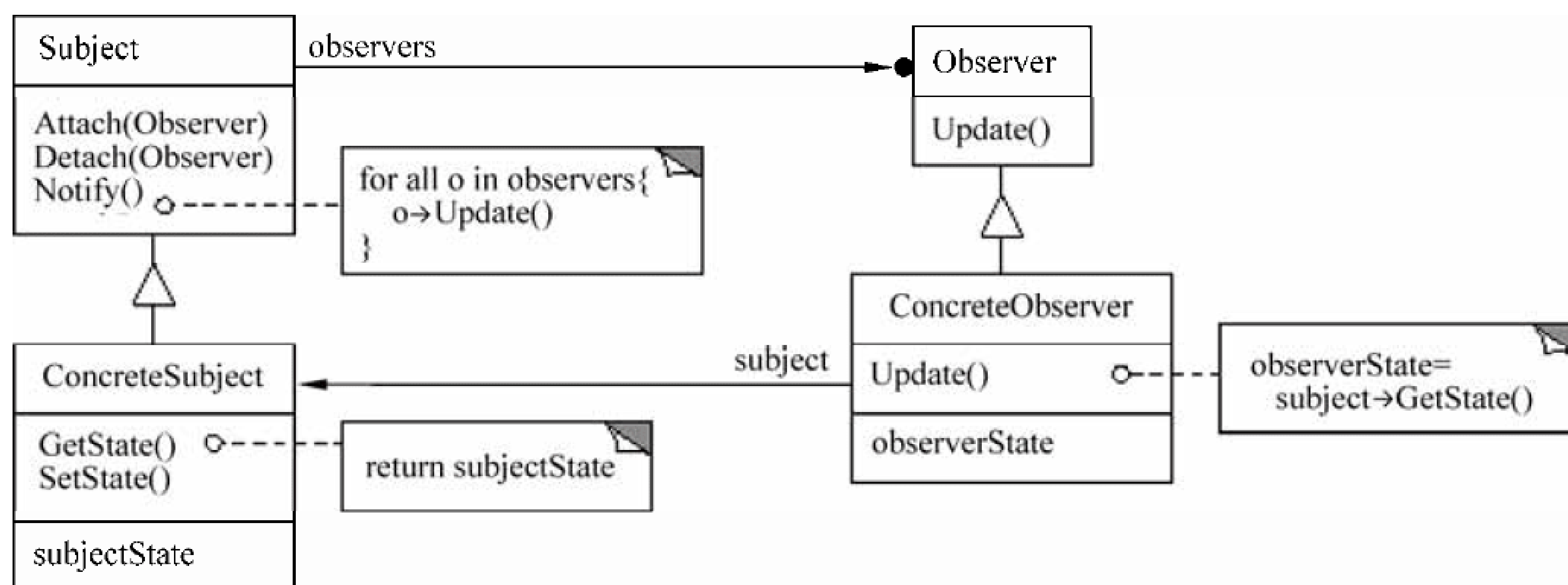
Observer 模式适用于：

① 当一个抽象模型有两个方面，其中一个方面依赖于另一个方面。将这两者封装在独立的对象中以使它们可以各自独立地改变和复用。

② 当对一个对象的改变需要同时改变其他对象，而不知道具体有多少对象有待改变时。

③ 当一个对象必须通知其他对象，而它又不能假定其他对象是谁，即：不希望这些对象是紧耦合的。

观察者模式的结构如下图所示，其中：



Subject（主题）知道它的观察者，可以有任意多个观察者观察同一个目标；提供注册和删除观察者对象的接口。

Observer（观察者）为那些在目标发生改变时需获得通知的对象定义一个更新接口。

ConcreteSubject（具体主题）将有关状态存入一个 ConcreteObserver 对象；当它的状态发生改变时，向它的各个观察者发出通知。

ConcreteObserver（具体观察者）维护一个指向 ConcreteSubject 对象的引用；存储有关状态，这些状态应与主题的状态保持一致；实现 Observer 的更新接口以使自身状态与主题的状态保持一致。

在本题的说明中，给出了观察者模式的结构图，答题时需要首先确定程序中的类与观察者模式结构的对应关系，也就是要找到哪些是 ConcreteSubject，哪些是 ConcreteObserver。由程序上下文可以判断出，类 EnvironmentData 对应的是 ConcreteSubject，类 CurrentConditionsDisplay 对应的是 ConcreteObserver。根据类图，它们分别为 Subject 和 Observer 的子类。由此可知，空（1）和空（5）应分别填写 Subject 和 Observer。

空（2）要求给出方法 notifyObservers 的实现，其功能是在主题发生变化时通知观察者。通知的实现是通过向对该主题感兴趣的所有观察者发送 update 消息实现的。对主题 EnvironmentData 感兴趣的观察者由向量 observers 表示，所以在 notifyObservers 方法中，就是对向量 observers 中的每个成员发送 update 消息，因此空（2）应填写

observer.update(temperature, humidity, cleanness)。

方法 measurementsChanged 表示主题发生了变化，这时应该通知对应的观察者，所以空（3）处应填写 notifyObservers()。

方法 setMeasurements 用于设置发生变化后的主题内容，所以空（4）处应填写 measurementsChanged()。

CurrentConditionsDisplay 是对主题 EnvironmentData 感兴趣的一个观察者，要能够获得主题的变化，需要首先将自己注册为该主题的观察者，这个注册行为在其构造函数中完成。因此空（6）处应填写 envData.registerObserver(this)。

试题六参考答案

（1）Subject

（2）observer.update(temperature, humidity, cleanness)

（3）notifyObservers()

（4）measurementsChanged()

（5）Observer

（6）envData.registerObserver(this)

第 23 章　2014 下半年软件设计师上午试题分析与解答

试题（1）

三总线结构的计算机总线系统由___（1）___组成。

（1）A．CPU 总线、内存总线和 IO 总线　　B．数据总线、地址总线和控制总线
C．系统总线、内部总线和外部总线　　D．串行总线、并行总线和 PCI 总线

试题（1）分析

本题考查计算机系统基础知识。

总线上传输的信息类型分为数据、地址和控制，因此总线由数据总线、地址总线和控制总线组成。

参考答案

（1）B

试题（2）

计算机采用分级存储体系的主要目的是为了解决___（2）___的问题。

（2）A．主存容量不足　　B．存储器读写可靠性
C．外设访问效率　　D．存储容量、成本和速度之间的矛盾

试题（2）分析

本题考查计算机系统基础知识。

计算机系统中，高速缓存一般用 SRAM，内存一般用 DRAM，外存一般采用磁存储器。SRAM 的集成度低、速度快、成本高。DRAM 的集成度高，但是需要动态刷新。磁存储器速度慢、容量大、价格便宜。因此，不同的存储设备组成分级存储体系，来解决速度、存储容量和成本之间的矛盾。

参考答案

（2）D

试题（3）

属于 CPU 中算术逻辑单元的部件是___（3）___。

（3）A．程序计数器　　B．加法器　　C．指令寄存器　　D．指令译码器

试题（3）分析

本题考查计算机系统基础知识。

程序计数器、指令寄存器和指令译码器都是 CPU 中控制单元的部件，加法器是算术逻辑运算单元的部件。

参考答案

（3）B

试题（4）

内存按字节编址从A5000H到DCFFFH的区域其存储容量为＿（4）＿。

（4）A．123KB　　B．180KB　　C．223KB　　D．224KB

试题（4）分析

本题考查计算机系统基础知识。

从地址A5000H到DCFFFH，存储单元数目为37FFFH（即224*1024）个，由于是字节编址，从而得到的存储容量为224KB。

参考答案

（4）D

试题（5）

以下关于RISC和CISC的叙述中，不正确的是＿（5）＿。

（5）A．RISC通常比CISC的指令系统更复杂

B．RISC通常会比CISC配置更多的寄存器

C．RISC编译器的子程序库通常要比CISC编译器的子程序库大得多

D．RISC比CISC更加适合VLSI工艺的规整性要求

试题（5）分析

本题考查计算机系统基础知识。

计算机工作时就是取指令和执行指令。一条指令往往可以完成一串运算的动作，但却需要多个时钟周期来执行。随着需求的不断增加，设计的指令集越来越多，为支持这些新增的指令，计算机的体系结构会越来越复杂，发展成CISC指令结构的计算机。而在CISC指令集的各种指令中，其使用频率却相差悬殊，大约有20%的指令会被反复使用，占整个程序代码的80%。而余下的80%的指令却不经常使用，在程序中常用的只占20%，显然这种结构是不太合理的。

RISC和CISC在架构上的不同主要有：

① 在指令集的设计上，RISC指令格式和长度通常是固定的（如ARM是32位的指令）、且寻址方式少而简单、大多数指令在一个周期内就可以执行完；CISC构架下的指令长度通常是可变的、指令类型也很多、一条指令通常要若干周期才可以执行完。由于指令集多少与复杂度上的差异，使RISC的处理器可以利用简单的硬件电路设计出指令解码功能，这样易于流水线的实现。相对的CISC则需要通过只读存储器里的微码来进行解码，CISC因为指令功能与指令参数变化较大，执行流水线作业时有较多的限制。

② RISC架构中只有载入和存储指令可以访问存储器，数据处理指令只对寄存器的内容进行操作。为了加速程序的运算，RISC会设定多组的寄存器，并且指定特殊用途的寄存器。CISC构架则允许数据处理指令对存储器进行操作，对寄存器的要求相对不高。

参考答案

（5）A

试题（6）

Flynn 分类法基于信息流特征将计算机分成 4 类，其中＿（6）＿只有理论意义而无实例。

（6）A．SISD　　B．MISD　　C．SIMD　　D．MIMD

试题（6）分析

本题考查计算机系统基础知识。

Flynn 主要根据指令流和数据流来分类，分为四类：

① 单指令流单数据流机器（SISD）

SISD 机器是一种传统的串行计算机，它的硬件不支持任何形式的并行计算，所有的指令都是串行执行，并且在某个时钟周期内，CPU 只能处理一个数据流。因此这种机器被称作单指令流单数据流机器。早期的计算机都是 SISD 机器。

② 单指令流多数据流机器（SIMD）

SIMD 是采用一个指令流处理多个数据流。这类机器在数字信号处理、图像处理以及多媒体信息处理等领域非常有效。

Intel 处理器实现的 MMXTM、SSE（Streaming SIMD Extensions）、SSE2 及 SSE3 扩展指令集，都能在单个时钟周期内处理多个数据单元。也就是说人们现在用的单核计算机基本上都属于 SIMD 机器。

③ 多指令流单数据流机器（MISD）

MISD 是采用多个指令流来处理单个数据流。在实际情况中，采用多指令流处理多数据流才是更有效的方法。因此 MISD 只是作为理论模型出现，没有投入实际应用。

④ 多指令流多数据流机器（MIMD）

MIMD 机器可以同时执行多个指令流，这些指令流分别对不同数据流进行操作。例如，Intel 和 AMD 的双核处理器就属于 MIMD 的范畴。

参考答案

（6）B

试题（7）

网络系统中，通常把＿（7）＿置于 DMZ 区。

（7）A．网络管理服务器　　B．Web 服务器
　　C．入侵检测服务器　　D．财务管理服务器

试题（7）分析

本题考查防火墙的基础知识。

DMZ 是指非军事化区，也称周边网络，可以位于防火墙之外也可以位于防火墙之内。非军事化区一般用来放置提供公共网络服务的设备，这些设备由于必须被公共网络

访问，所以无法提供与内部网络主机相等的安全性。

分析四个备选答案，Web 服务器是为一种为公共网络提供 Web 访问的服务器；网络管理服务器和入侵检测服务器是管理企业内部网和对企业内部网络中的数据流进行分析的专用设备，一般不对外提供访问；而财务服务器是一种仅针对财务部门内部访问和提供服务的设备，不提供对外的公共服务。

参考答案

（7）B

试题（8）

以下关于拒绝服务攻击的叙述中，不正确的是__（8）__。

（8）A．拒绝服务攻击的目的是使计算机或者网络无法提供正常的服务

B．拒绝服务攻击是不断向计算机发起请求来实现的

C．拒绝服务攻击会造成用户密码的泄漏

D．DDoS 是一种拒绝服务攻击形式

试题（8）分析

本题考查拒绝服务攻击的基础知识。

拒绝服务攻击是指不断对网络服务系统进行干扰，改变其正常的作业流程，执行无关程序使系统响应减慢直至瘫痪，从而影响正常用户的使用。当网络服务系统响应速度减慢或者瘫痪时，合法用户的正常请求将不被响应，从而实现用户不能进入计算机网络系统或不能得到相应的服务的目的。

DDoS 是分布式拒绝服务的英文缩写。分布式拒绝服务的攻击方式是通过远程控制大量的主机向目标主机发送大量的干扰消息的一种攻击方式。

参考答案

（8）C

试题（9）

__（9）__不是蠕虫病毒。

（9）A．熊猫烧香　　B．红色代码　　C．冰河　　D．爱虫病毒

试题（9）分析

本题考查计算机病毒的基础知识。

“蠕虫”（Worm）是一个程序或程序序列。它利用网络进行复制和传播，传染途径是通过网络、移动存储设备和电子邮件。最初的蠕虫病毒定义是在 DOS 环境下，病毒发作时会在屏幕上出现一条类似虫子的东西，胡乱吞吃屏幕上的字母并将其改形，蠕虫病毒因此而得名。常见的蠕虫病毒有红色代码、爱虫病毒、熊猫烧香、Nimda 病毒、爱丽兹病毒等。

冰河是木马软件，主要用于远程监控，冰河木马后经其他人多次改写形成多种变种，并被用于入侵其他用户的计算机的木马程序。

参考答案

（9）C

试题（10）

甲公司接受乙公司委托开发了一项应用软件，双方没有订立任何书面合同。在此情形下，＿（10）＿享有该软件的著作权。

（10）A．甲公司　　B．甲、乙公司共同
C．乙公司　　D．甲、乙公司均不

试题（10）分析

委托开发软件著作权关系的建立，通常由委托方与受委托方订立合同而成立。委托开发软件关系中，委托方的责任主要是提供资金、设备等物质条件，并不直接参与开发软件的创作开发活动。受托方的主要责任是根据委托合同规定的目标开发出符合条件的软件。关于委托开发软件著作权的归属，《计算机软件保护条例》第十二条规定："受他人委托开发的软件，其著作权的归属由委托者与受委托者签定书面协议约定，如无书面协议或者在协议中未作明确约定，其著作权属于受委托者。"根据该条的规定，确定委托开发的软件著作权的归属应当掌握两条标准：

① 委托开发软件系根据委托方的要求，由委托方与受托方以合同确定的权利和义务的关系而进行开发的软件，因此软件著作权归属应当作为合同的重要条款予以明确约定。对于当事人已经在合同中约定软件著作权归属关系的，如事后发生纠纷，软件著作权的归属仍应当根据委托开发软件的合同来确定。

② 对于在委托开发软件活动中，委托者与受委托者没有签定书面协议，或者在协议中未对软件著作权归属作出明确的约定，其软件著作权属于受委托者，即属于实际完成软件的开发者。

参考答案

（10）A

试题（11）

甲、乙软件公司于 2013 年 9 月 12 日就其财务软件产品分别申请"大堂"和"大唐"商标注册。两财务软件相似，且经协商双方均不同意放弃使用其申请注册的商标标识。此情形下，＿（11）＿获准注册。

（11）A．"大堂"　　B．"大堂"与"大唐"都能
C．"大唐"　　D．由甲、乙抽签结果确定谁能

试题（11）分析

我国商标注册采取"申请在先"的审查原则，当两个或两个以上申请人在同一种或者类似商品上申请注册相同或者近似商标时，商标主管机关根据申请时间的先后，决定商标权的归属，申请在先的人可以获得注册。对于同日申请的情况，使用在先的人可以获得注册。如果同日使用或均未使用，则采取申请人之间协商解决，协商不成的，由各

申请人抽签决定。

类似商标是指在同一种或类似商品上用作商标的文字、图形、读音、含义或文字与图形的整体结构上等要素大体相同的商标，即易使消费者对商品的来源产生误认的商标。甲、乙两公司申请注册的商标，“大堂”与“大唐”读音相同、文字相近似，不能同时获准注册。在协商不成的情形下，由甲、乙公司抽签结果确定谁能获准注册。

参考答案

（11）D

试题（12）、（13）

以下媒体中，___（12）___是表示媒体，___（13）___是表现媒体。

（12）A. 声音　B. 声音编码　C. 超声波　D. 喇叭

（13）A. 声音　B. 声音编码　C. 超声波　D. 喇叭

试题（12）、（13）

本题考查多媒体基础知识。传输媒体指传输表示媒体的物理介质，如电缆、光缆、电磁波等；表示媒体指传输感觉媒体，如声音、图像等的中介媒体，即用于数据交换的编码，如文本编码、声音编码和图像编码等；表现媒体是指进行信息输入和输出的媒体，如键盘、鼠标、话筒以及显示器、打印机、喇叭等；存储媒体指用于存储表示媒体的物理介质，如硬盘、光盘等。

声音信号的两个基本参数是幅度和频率。幅度是指声波的振幅，通常用动态范围表示，一般用分贝（dB）为单位来计量。频率是指声波每秒钟变化的次数，用 Hz 表示。人们把频率小于 20Hz 声波信号称为亚音信号（也称次音信号）；频率范围为 20Hz～20kHz 的声波信号称为音频信号；高于 20kHz 的信号称为超音频信号（也称超声波）。

参考答案

（12）B　（13）D

试题（14）

显示深度、图像深度是图像显示的重要指标。当___（14）___时，显示器不能完全反映数字图像中使用的全部颜色。

（14）A. 显示深度＝图像深度　B. 显示深度＞图像深度

C. 显示深度≥图像深度　D. 显示深度＜图像深度

试题（14）分析

本题考查考生多媒体基础知识。

图像深度是指存储每个像素所用的位数，它是用来度量图像的色彩分辨率的，即确定彩色图像的每个像素可能有的颜色数，或者确定灰度图像的每个像素可能有的灰度级数。显示深度是显示器上每个点用于显示颜色的二进制位数。使用显示器显示数字图像时，应当使显示器的显示深度大于或等于数字图像的深度，这样显示器就可以完全反映数字图像中使用的全部颜色。

参考答案

（14）D

试题（15）

以下关于结构化开发方法的叙述中，不正确的是__（15）__。

（15）A. 总的指导思想是自顶向下，逐层分解

B. 基本原则是功能的分解与抽象

C. 与面向对象开发方法相比，更适合于大规模、特别复杂的项目

D. 特别适合于数据处理领域的项目

试题（15）分析

本题考查结构化开发方法的基础知识。

结构化开发方法由结构化分析、结构化设计和结构化程序设计构成，是一种面向数据流的开发方法。结构化方法总的指导思想是自顶向下、逐层分解，基本原则是功能的分解与抽象。它是软件工程中最早出现的开发方法，特别适合于数据处理领域的问题，但是不适合解决大规模的、特别复杂的项目，而且难以适应需求的变化。

参考答案

（15）C

试题（16）

模块 A、B 和 C 都包含相同的 5 个语句，这些语句之间没有联系。为了避免重复，把这 5 个语句抽取出来组成一个模块 D，则模块 D 的内聚类型为__（16）__内聚。

（16）A. 功能 B. 通信 C. 逻辑 D. 巧合

试题（16）分析

本题考查软件设计的相关知识。

模块独立性是创建良好设计的一个重要原则，一般采用模块间的耦合和模块的内聚两个准则来进行度量。内聚是指模块内部各元素之间联系的紧密程度，内聚度越高，则模块的独立性越好。内聚性一般有以下几种：

① 巧合内聚，指一个模块内的各处理元素之间没有任何联系。

② 逻辑内聚，指模块内执行几个逻辑上相似的功能，通过参数确定该模块完成哪一个功能。

③ 时间内聚，把需要同时执行的动作组合在一起形成的模块。

④ 通信内聚，指模块内所有处理元素都在同一个数据结构上操作，或者指各处理使用相同的输入数据或者产生相同的输出数据。

⑤ 顺序内聚，指一个模块中各个处理元素都密切相关于同一功能且必须顺序执行，前一个功能元素的输出就是下一个功能元素的输入。

⑥ 功能内聚是最强的内聚，指模块内所有元素共同完成一个功能，缺一不可。

参考答案

（16）D

试题（17）

某个项目在开发时采用了不成熟的前沿技术，由此而带来的风险属于__（17）__风险。

（17）A．市场　　B．技术　　C．经济　　D．商业

试题（17）分析

本题考查软件开发风险的基本概念。

风险是一种具有负面后果的、人们不希望发生的事件。从不同的角度可以对风险进行不同的分类。如从风险涉及的范围，风险可以分为项目风险、技术风险和商业风险等。技术风险涉及设计方案、实现、接口、验证以及维护等方面的问题。此外，包括需求规格说明的不确定性、技术的不确定性、技术的陈旧以及采用不成熟的前沿技术等可能会带来技术风险。技术风险威胁着开发产品的质量和交付产品的时间。

参考答案

（17）B

试题（18）

属于面向对象、解释型程序设计语言的是__（18）__。

（18）A．XML　　B．Python　　C．Prolog　　D．C++

试题（18）分析

本题考查程序语言基础知识。

XML（可扩展标记语言）是标准通用标记语言的子集，是一种用于标记电子文件使其具有结构性的标记语言。

Python 是一种面向对象、解释型计算机程序设计语言。

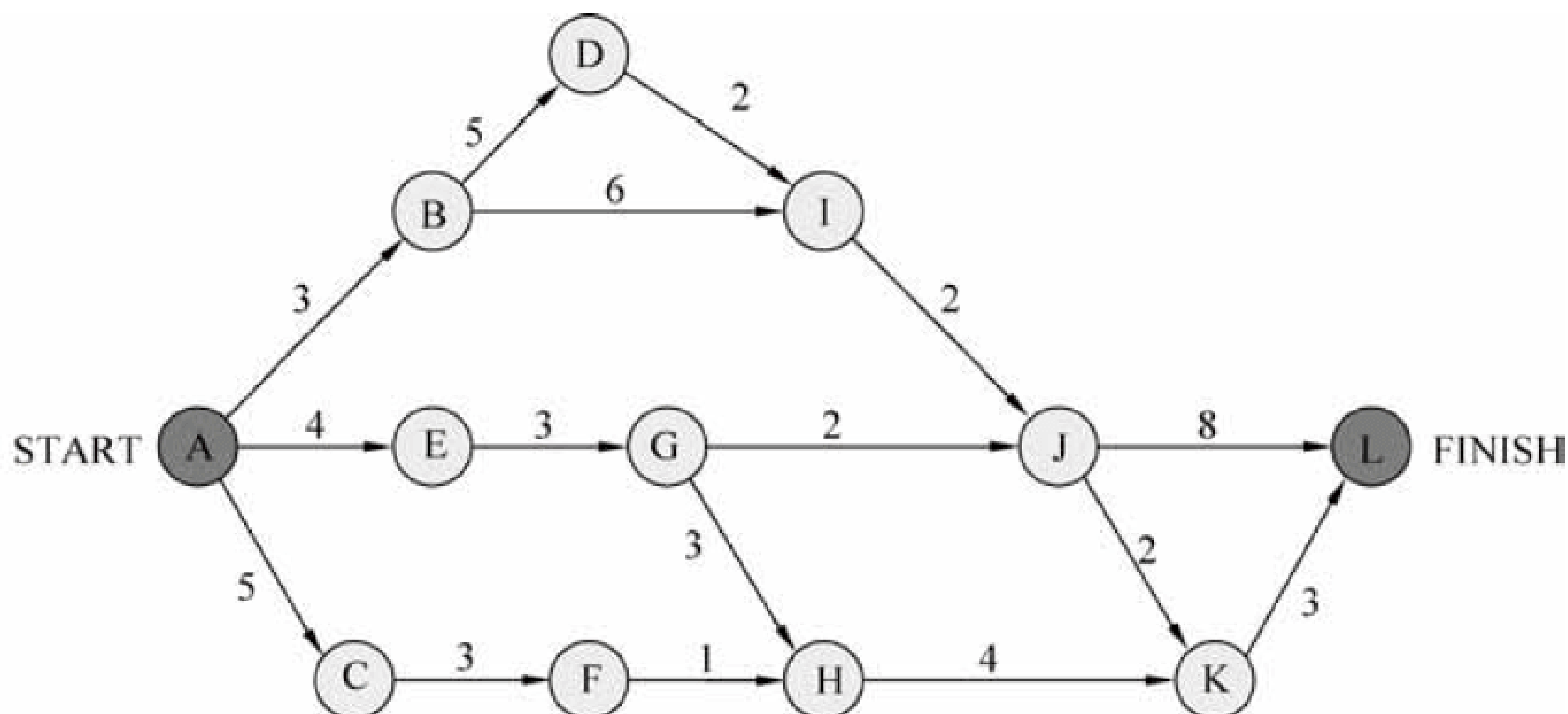
Prolog 是逻辑型程序设计语言。

参考答案

（18）B

试题（19）、（20）

下图是一个软件项目的活动图，其中顶点表示项目里程碑，连接顶点的边表示活动，边的权重表示活动的持续时间，则里程碑__（19）__在关键路径上。活动 GH 的松弛时间是__（20）__。



（19）A．B　　B．E　　C．C　　D．K

（20）A．0　　B．1　　C．2　　D．3

试题（19）、（20）分析

本题考查活动图的基础知识。

根据关键路径法，计算出关键路径为 A—B—D—I—J—L，关键路径长度为 20，因此里程碑 B 在关键路径上，而里程碑 E、C 和 K 不在关键路径上。包含活动 GH 的最长路径是 A—E—G—H—K—L，长度为 17，因此该活动的松弛时间为 20–17=3。

参考答案

（19）A　　（20）D

试题（21）

算术表达式“(a-b)*(c+d)”的后缀式是___（21）___。

（21）A．ab-cd+*　　B．abcd-*+　　C．ab-*cd+　　D．ab-c+d*

试题（21）分析

本题考查程序语言基础知识。

后缀式（逆波兰式）是波兰逻辑学家卢卡西维奇发明的一种表示表达式的方法。这种表示方式把运算符写在运算对象的后面，例如把 a+b 写成 ab+，所以也称为后缀式。算术表达式“(a-b)*(c+d)”的后缀式是“ab-cd+*”。

参考答案

（21）A

试题（22）

将高级语言源程序翻译成机器语言程序的过程中，常引入中间代码。以下关于中间代码的叙述中，不正确的是___（22）___。

（22）A．中间代码不依赖于具体的机器

B．使用中间代码可提高编译程序的可移植性

C．中间代码可以用树或图表示

D．中间代码可以用栈和队列表示

试题（22）分析

本题考查程序语言基础知识。

从原理上讲，对源程序进行语义分析之后就可以直接生成目标代码，但由于源程序与目标代码的逻辑结构往往差别很大，特别是考虑到具体机器指令系统的特点，要使翻译一次到位很困难，而且用语法制导方式机械生成的目标代码往往是烦琐和低效的，因此有必要设计一种中间代码，将源程序首先翻译成中间代码表示形式，以利于进行与机器无关的优化处理。由于中间代码实际上也起着编译器前端和后端分水岭的作用，所以使用中间代码也有助于提高编译程序的可移植性。常用的中间代码有后缀式、三元式、四元式和树（图）等形式。

参考答案

（22）D

试题（23）

假设系统采用PV操作实现进程同步与互斥。若n个进程共享两台打印机，那么信号量S的取值范围为＿（23）＿。

（23）A．-2～n　　B．-(n-1)～1　　C．-(n-1)～2　　D．- (n-2)～2

试题（23）分析

本题考查操作系统PV操作方面的基本知识。

系统采用PV操作实现进程同步与互斥，若有n个进程共享两台打印机，那么信号量S初值应为2。当第1个进程执行P(S)操作时，信号量S的值减去1后等于1；当第2个进程执行P(S)操作时，信号量S的值减去1后等于0；当第3个进程执行P(S)操作时，信号量S的值减去1后等于-1；当第4个进程执行P(S)操作时，信号量S的值减去1后等于-2；……；当第n个进程执行P(S)操作时，信号量S的值减去1后等于-(n-2)。可见，信号量S的取值范围为-(n-2)～2。

参考答案

（23）D

试题（24）

假设段页式存储管理系统中的地址结构如下图所示，则系统＿（24）＿。

31　　　　　　22	21　　　　　　12	11　　　　　　0
段　号	页　号	页内地址

（24）A．最多可有2048个段，每个段的大小均为2048个页，页的大小为2K

B．最多可有2048个段，每个段最大允许有2048个页，页的大小为2K

C．最多可有1024个段，每个段的大小均为1024个页，页的大小为4K

D．最多可有1024个段，每个段最大允许有1024个页，页的大小为4K

试题（24）分析

本题考查操作系统页式存储管理方面的基础知识。从图中可见，页内地址的长度是12位，2^{12}=4096，即4K；页号部分的地址长度是10位，每个段最大允许有2^{10}=1024个页；段号部分的地址长度是10位，2^{10}=1024，最多可有1024个段。故本题的正确答案为D。

参考答案

（24）　D

试题（25）、（26）

假设磁盘块与缓冲区大小相同，每个盘块读入缓冲区的时间为10μs，由缓冲区送至用户区的时间是5μs，系统对每个磁盘块数据的处理时间为2μs。若用户需要将大小为

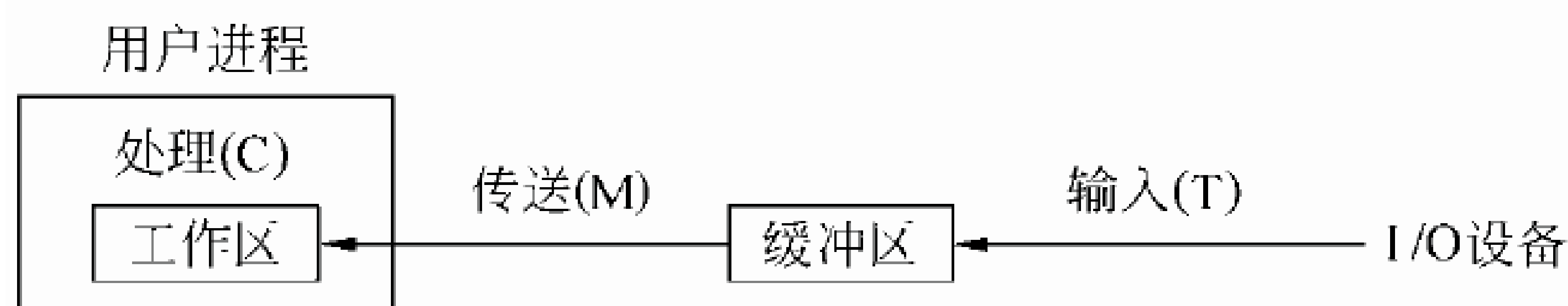
10 个磁盘块的 Doc1 文件逐块从磁盘读入缓冲区，并送至用户区进行处理，那么采用单缓冲区需要花费的时间为__（25）__μs；采用双缓冲区需要花费的时间为__（26）__μs。

（25）A. 100　　B. 107　　C. 152　　D. 170

（26）A. 100　　B. 107　　C. 152　　D. 170

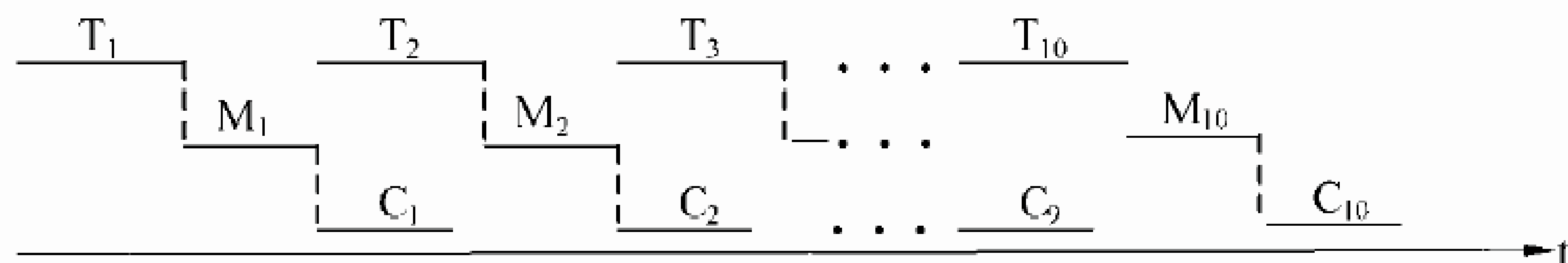
试题（25）、（26）分析

试题（25）的正确的答案为 C。在块设备输入时，假定从磁盘把一块数据输入到缓冲区的时间为 T，缓冲区中的数据传送到用户工作区的时间为 M，而系统处理（计算）的时间为 C，如图（a）所示。



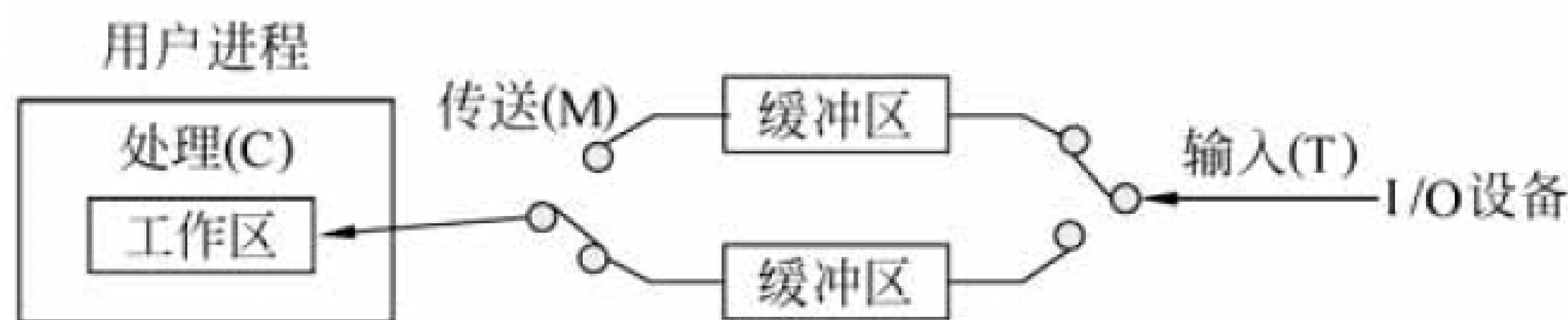
图（a）　单缓冲工作过程图

当第一块数据送入用户工作区后，缓冲区是空闲的，可以传送第二块数据。这样第一块数据的处理 C1 与第二块数据的输入 T2 是可以并行的，如图（b）所示，依次类推。系统对每一块数据的处理时间为：Max(C,T)+M。因为当 T>C 时，处理时间为 M+T；当 T<C 时，处理时间为 M+C。本题每一块数据的处理时间为 10+5=15，Doc1 文件的处理时间为 15*10+2。



图（b）　单缓冲并行工作示意图

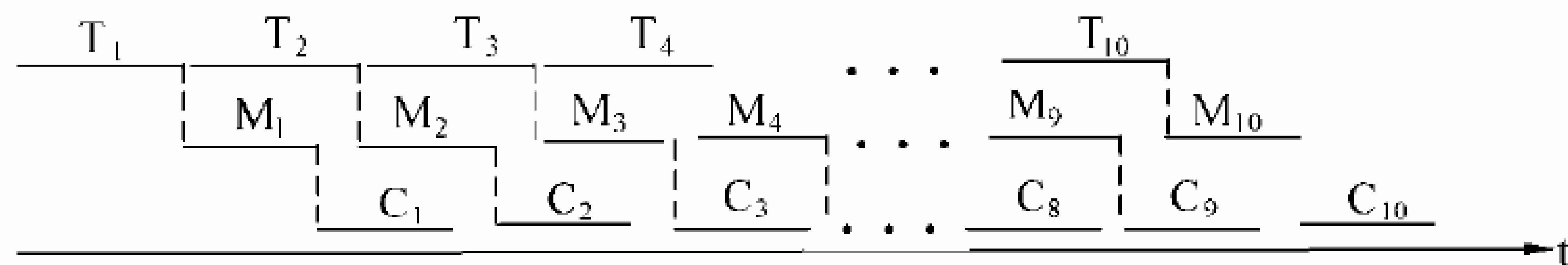
试题（26）的正确答案为 B。双缓冲工作方式基本方法是在设备输入时，先将数据输入到缓冲区 1，装满后便转向缓冲区 2。此时系统可以从缓冲区 1 中提取数据传送到用户区，最后由系统对数据进行处理，如图（c）所示。



图（c）　双缓冲工作过程图

双缓冲可以实现对缓冲区中数据的输入 T 和提取 M，与 CPU 的计算 C，三者并行工作，如图（c）所示。从图中可以看出，双缓冲进一步加快了 I/O 的速度，提高了设备的利用率。在双缓冲时，系统处理一块数据的时间可以粗略地认为是 Max(C,T)。如果 C<T，可使块设备连续输入；如果 C>T，则可使系统不必等待设备输入。本题每一块数

据的处理时间为 10，采用双缓冲需要花费的时间为 10*10+5+2=107。



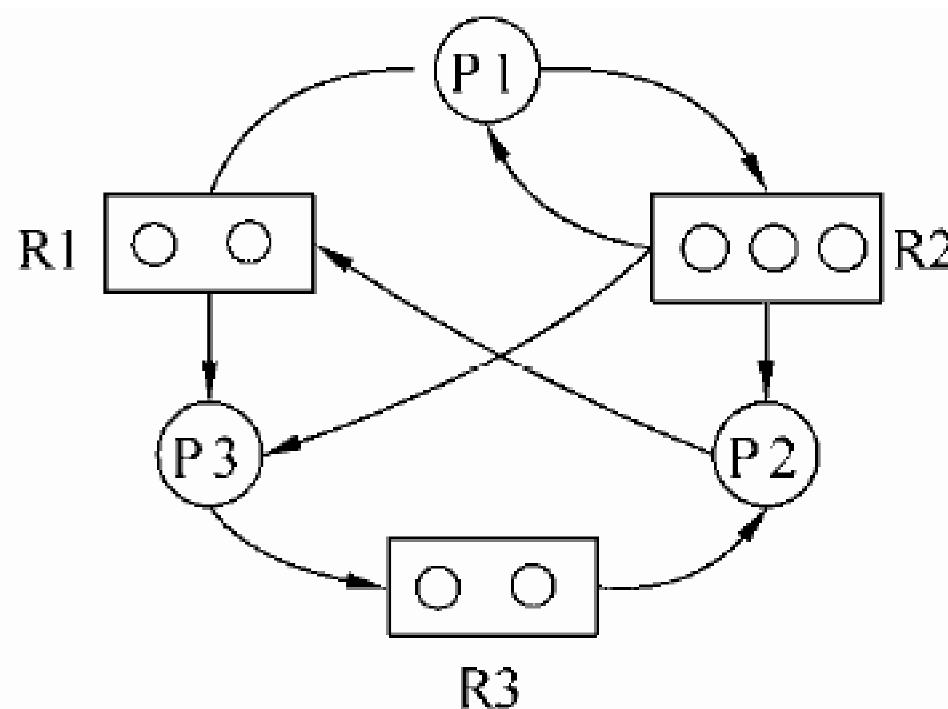
图（d） 双缓冲并行工作示意图

参考答案

（25）C　　（26）B

试题（27）、（28）

在如下所示的进程资源图中，__（27）__；该进程资源图是__（28）__。



（27）A．P1、P2、P3 都是阻塞节点

B．P1 是阻塞节点、P2、P3 是非阻塞节点

C．P1、P2 是阻塞节点、P3 是非阻塞节点

D．P1、P2 是非阻塞节点、P3 是阻塞节点

（28）A．可以化简的，其化简顺序为 P1→P2→P3

B．可以化简的，其化简顺序为 P3→P1→P2

C．可以化简的，其化简顺序为 P2→P1→P3

D．不可以化简的，因为 P1、P2、P3 申请的资源都不能得到满足

试题（27）、（28）分析

试题（27）的正确选项为 C。图中 R1 资源只有 2 个，P2 进程申请该资源得不到满足，故 P2 进程是阻塞节点； R2 资源只有 3 个，P1 申请该资源得不到满足，故 P1 进程也是阻塞节点；R3 资源只有 2 个，分配给 P1 进程 1 个，P3 申请 1 个该资源可以得到满足，故 P3 是非阻塞节点。

试题（28）的正确选项为 B。R3 资源有 2 个，已分配 1 个，P3 申请 1 个 R2 资源可以得到满足，故进程 P3 可以运行完毕释放其占有的资源。这样可以使得 P1、P2 都变为非阻塞节点，得到所需资源运行完毕，因此，进程资源图是可化简的。其化简顺序可以为 P3→P1→P2 或 P3→P2→P1。

参考答案

（27）C　（28）B

试题（29）

以下关于增量模型的叙述中，正确的是__（29）__。

（29）A. 需求被清晰定义

B. 可以快速构造核心产品

C. 每个增量必须要进行风险评估

D. 不适宜商业产品的开发

试题（29）分析

本题考查软件开发过程的基础知识。

软件开发过程以系统需求作为输入，以要交付的产品作为输出，涉及活动、约束和资源使用的一系列工具和技术。瀑布模型、快速原型化模型、增量模型、螺旋模型等都是典型的软件开发过程模型。增量模型是 Mills 等于 1980 年提出来的。在使用该模型开发软件时，把软件产品作为一系列的增量构件来设计、编码、集成和测试。每个构件由多个相互作用的模块构成，并能够完成特定的功能。

其优点包括能在较短时间内向用户提交可完成一些有用的工作产品；用户有充裕的时间来学习和适应不断增加的产品功能；项目失败风险较低；优先级最高的服务首先交付，然后再逐步增加新的构件，这样最重要的构件被测试得最充分。在四个选项中，只有选项 B 是描述增量模型的，要求需求被清晰定义是瀑布模型的一个典型特点，风险评估是螺旋模型的特点。在当今市场竞争激烈的条件下，用增量模型可以快速地交付一部分产品，是适于商业产品的开发的。

参考答案

（29）B

试题（30）

以下关于 CMM 的叙述中，不正确的是__（30）__。

（30）A. CMM 是指软件过程能力成熟度模型

B. CMM 根据软件过程的不同成熟度划分了 5 个等级，其中，1 级被认为成熟度最高，5 级被认为成熟度最低

C. CMMI 的任务是将已有的几个 CMM 模型结合在一起，使之构造成为“集成模型”

D. 采用更成熟的 CMM 模型，一般来说可以提高最终产品的质量

试题（30）分析

本题考查软件过程的基础知识。

CMM（Capability Maturity Model）是指软件过程能力成熟度模型，该模型按照软件过程的不同成熟度划分了 5 个等级，1 级被认为成熟度最低，5 级则为成熟度最高。一般

来说，采用更成熟的软件过程模型，往往可以得到更高质量的软件产品。1997年美国卡内基·梅隆大学软件工程研究所SEI将已有的几个CMM模型结合在一起，构造成“集成模型”即CMMI（Capability Maturity Model Integration）。

参考答案

（30）B

试题（31）

在ISO/IEC软件质量模型中，可靠性是指在规定的一段时间内和规定的条件下，软件维持在其性能水平的能力，其子特性不包括__（31）__。

（31）A. 成熟性 B. 容错性 C. 易恢复 D. 可移植性

试题（31）分析

本题考查软件质量的基础知识。

ISO/IEC9126软件质量模型由三个层次组成：第一层是质量特性，第二层是质量子特性，第三层是度量指标。可靠性是一个重要的质量特性，其子特性包括成熟性、容错性和易恢复性。

参考答案

（31）D

试题（32）

在软件开发过程中，系统测试阶段的测试目标来自于__（32）__阶段。

（32）A. 需求分析 B. 概要设计 C. 详细设计 D. 软件实现

试题（32）分析

本题考查软件测试的基础知识。

软件测试的基本目标是为了发现软件中的错误，但软件测试分为几个不同的阶段，每个阶段的侧重点是有所不同的。单元测试主要是发现程序代码中的问题，针对详细设计和软件实现阶段的工作进行的；集成测试验证系统模块是否能够根据系统和程序设计规格说明的描述进行工作，即模块以及模块之间的接口的测试；而系统测试则是验证系统是否确实执行需求规格说明中描述的功能和非功能要求，因此测试目标在需求分析阶段就已经定义。

参考答案

（32）A

试题（33）

以下关于文档的叙述中，不正确的是__（33）__。

（33）A. 项目相关人员可以通过文档进行沟通

B. 编写文档会降低软件开发的效率

C. 编写高质量文档可以提高软件开发的质量

D. 文档是软件的不可或缺的部分

试题（33）分析

本题考查软件文档的相关知识。

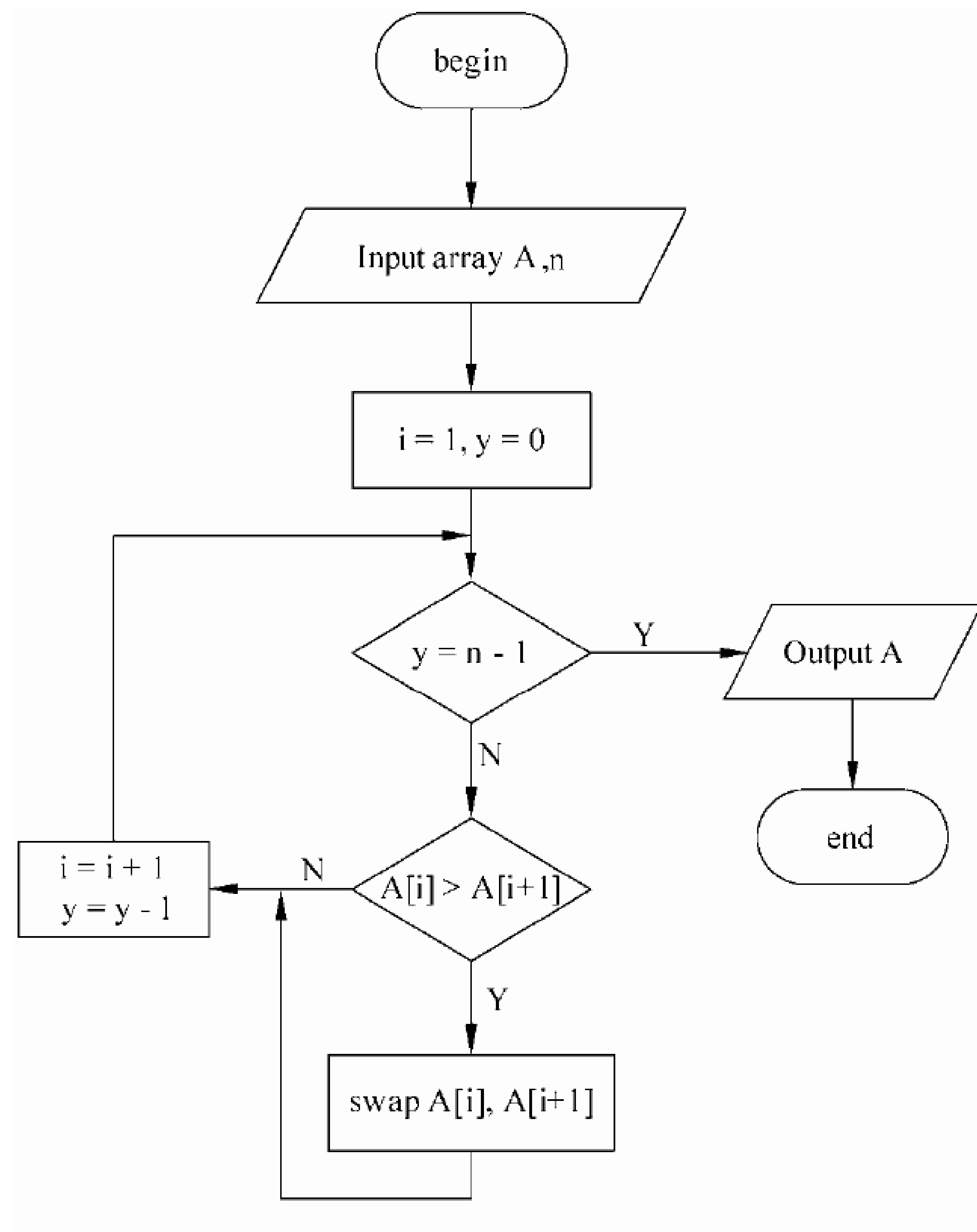
软件由程序、数据和相关文档构成，因此文档是软件不可或缺的重要组成部分。软件开发各个阶段都需要撰写相关文档，如开发计划、需求分析文档、设计文档等，这些文档是开发人员之间以及和其他人员之间进行沟通的重要依据，高质量的文档对于提高软件开发质量具有重要的意义。尽管在开发过程中编写文档需要占用开发时间，但是相对于没有文档而言，编写文档使得开发人员对各个阶段的工作都进行周密思考，全盘权衡，从而减少返工。并且可以在开发早期发现错误和不一致性，便于及时加以纠正，因此可以提高软件开发效率。

参考答案

（33）B

试题（34）、（35）

下图所示的程序流程图中有___（34）___条不同的简单路径。采用 McCabe 度量法计算该程序图的环路复杂性为___（35）___。



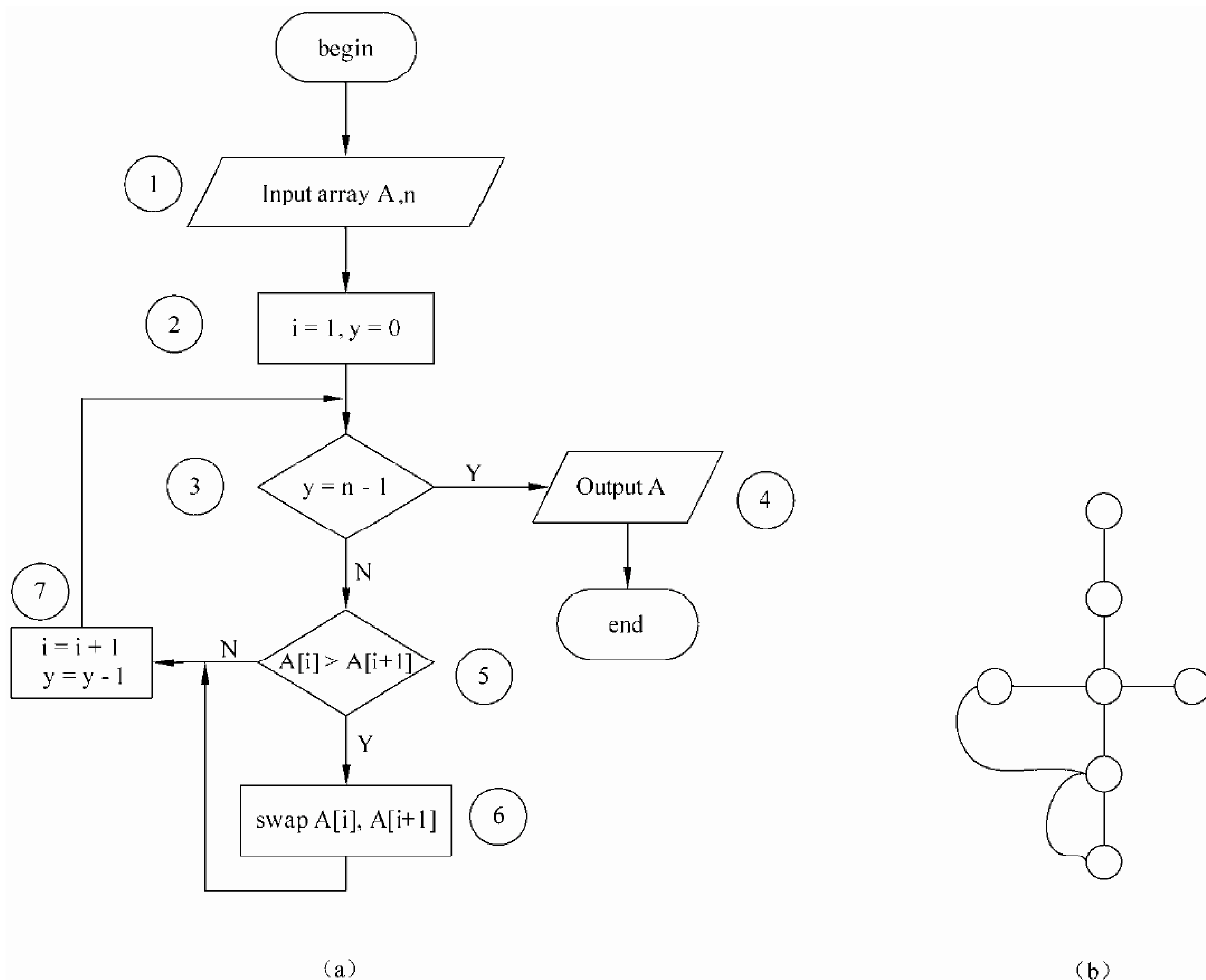
（34）A．3　　B．4　　C．5　　D．6

（35）A．3　　B．4　　C．5　　D．6

试题（34）、（35）分析

本题考查程序结构和复杂性的基本概念和相关知识。

重新画出上述流程图，给相关的操作加上编号，并给出对应的等价图，如下图（a）和（b）所示。



从图（a）可知，共有3条简单路径，即①②③④；①②③⑤⑦③；①②③⑤⑥⑦③。根据图（b）计算程序复杂性，得到n=7，e=8，因此复杂性为e-n+2=3。

参考答案

（34）A　　（35）A

试题（36）

以下关于软件维护和可维护性的叙述中，不正确的是__（36）__。

（36）A．软件维护要解决软件产品交付用户之后运行中发生的各种问题

B．软件的维护期通常比开发期长得多，其投入也大得多

C．进行质量保证审查可以提高软件产品的可维护性

D．提高可维护性是在软件维护阶段考虑的问题

试题（36）分析

本题考查软件维护的相关知识。

软件产品在交付给用户之后，就进入了维护阶段。在该阶段针对系统改变所做的任何工作，都属于维护活动。软件维护期通常比开发期要长得多，根据统计数据一般项目花费 1 到 2 年的开发时间，但是需要额外的 5 到 6 年的维护时间。维护的成本也比开发成本高得多，另外一些企业数据表明，平均 39%的工作量花在开发上，其余的在维护上。软件可维护性是指在给定的条件下，在规定的时间间隔内，使用规定的过程和资源完成维护活动的概率。通过多个方面来提高软件产品的可维护性，其中进行质量保证审查是一个重要的手段。在软件开发的各个阶段都需要考虑提高软件产品的可维护性，而不仅仅是在软件维护阶段。

参考答案

（36）D

试题（37）

类 （37） 之间存在着一般和特殊的关系。

（37）A．汽车与轮船　　B．交通工具与飞机
　　　C．轮船与飞机　　D．汽车与飞机

试题（37）分析

本题考查面向对象的基本知识。

在进行类设计时，有些类之间存在一般和特殊关系，即一些类是某个类的特殊情况，某个类是一些类的一般情况，这就是继承关系。继承是类之间的一种关系，在定义和实现一个类的时候，可以在一个已经存在的类（一般情况）的基础上来进行，把这个已经存在的类所定义的内容作为自己的内容，并加入若干新的内容，即子类比父类更加具体化。交通工具是泛指各类交通工具，而汽车、飞机和轮船分别都是具体的交通工具类，且具有自己的特性。因此交通工具是汽车、飞机和轮船类的一般情况，分别与汽车、轮船和飞机存在一般与特殊的关系。

参考答案

（37）B

试题（38）

多态分为参数多态、包含多态、过载多态和强制多态四种不同形式，其中 （38） 多态在许多语言中都存在，最常见的例子就是子类型化。

（38）A．参数　　B．包含　　C．过载　　D．强制

试题（38）分析

本题考查面向对象的基本知识。

面向对象系统中，在收到消息时，对象要予以相应。多态（polymorphism）是不同的对象收到同一消息可以进行不同的响应，产生完全不同的结果，用户可以发送一个通

用的消息，而实现细节则由接收对象自行决定，使得同一个消息就可以调用不同的方法，即一个对象具有多种形态。Cardelli 和 Wegner 将多态分为 4 种不同的形式：参数多态、包含多态、过载多态和强制多态。其中参数多态是应用比较广的多态，包含多态在许多语言中都存在，最常见的例子就是子类型化。过载多态是同一个名字在不同的上下文中所代表的含义。

参考答案

（38）B

试题（39）、（40）

在面向对象程序设计语言中，对象之间通过__（39）__方式进行通信。以下关于好的面向对象程序设计语言的叙述中，不正确的是__（40）__。

（39）A. 消息传递 B. 继承 C. 引用 D. 多态

（40）A. 应该支持被封装的对象 B. 应该支持类与实例的概念
C. 应该支持通过指针进行引用 D. 应该支持继承和多态

试题（39）、（40）分析

本题考查面向对象的基本知识。

在面向对象技术中，继承关系是一种模仿现实世界中继承关系的一种类之间的关系，是超类（父类）和子类之间共享数据和方法的机制。面向对象语言中对象通过消息传递的方式进行相互通信。在继承的支持下，不同的对象收到同一消息可以进行不同的响应，并且会产生完全不同的结果，这种现象称为多态。

好的面向对象程序设计语言一般应该支持被封装的对象、类与实例的概念、支持继承和多态等面向对象技术中的概念。

参考答案

（39）A （40）C

试题（41）、（42）

UML 中有 4 种事物：结构事物、行为事物、分组事物和注释事物。类、接口、构件属于__（41）__事物；一个依附于一个元素或一组元素之上对其进行约束或解释的简单符号为__（42）__事物。

（41）A. 结构 B. 行为 C. 分组 D. 注释

（42）A. 结构 B. 行为 C. 分组 D. 注释

试题（41）、（42）分析

本题考查统一建模语言（UML）的基本知识。

UML 是面向对象软件的标准化建模语言，由三个要素构成：UML 的基本构造块、支配这些构造块如何放置在一起的规则和运用与整个语言的一些公共机制。UML 的词汇表包含三种构造块：事物、关系和图。事物是对模型中最具有代表性的成分的抽象；关系把事物结合在一起；图聚集了相关的事物。

UML 中有 4 种事物：结构事物、行为事物、分组事物和注释事物。结构事物是 UML 模型中的名词，它们通常是模型的静态部分，描述概念或物理元素。结构事物包括类（class）、接口（interface）、协作（collaboration）、用例（use case）、主动类（active class）、构件（component）、制品（artifact）和结点（node）。行为事物是 UML 模型的动态部分，它们是模型中的动词，描述了跨越时间和空间的行为。行为事物包括：交互（interaction）、状态机（state machine）和活动（activity）。分组事物是 UML 模型的组织部分，是一些由模型分解成的“盒子”。在所有的分组事物中最主要的分组事物是包（package）。注释事物是 UML 模型的解释部分。这些注释事物用来描述、说明和标注模型的任何元素。注解（note）是一种主要的注释事物。注解是一个依附于一个元素或者一组元素之上，对它进行约束或解释的简单符号。

参考答案

（41）A　　　（42）D

试题（43）、（44）

一组对象以定义良好但是复杂的方式进行通信，产生的相互依赖关系结构混乱且难以理解。采用＿（43）＿模式，用一个中介对象来封装一系列的对象交互，从而使各对象不需要显式地相互引用，使其耦合松散，而且可以独立地改变它们之间的交互。此模式与＿（44）＿模式是相互竞争的模式，主要差别是：前者的中介对象封装了其他对象间的通信，而后者通过引入其他对象来分布通信。

（43）A．解释器（Interpreter）　　B．策略（Strategy）
　　　C．中介者（Mediator）　　D．观察者（Observer）

（44）A．解释器（Interpreter）　　B．策略（Strategy）
　　　C．中介者（Mediator）　　D．观察者（Observer）

试题（43）、（44）分析

本题考查设计模式的基本概念。

解释器（Interpreter）设计模式是给定一个语言，定义它的文法的一种表示，并定义一个解释器，这个解释器使用该表示来解释语言中的句子。策略（Strategy）设计模式定义一系列算法，把它们一个个封装起来，并且使它们可相互替换。这一模式使得算法可独立于它的客户而变化。中介者（Mediator）用一个中介对象来封装一系列的对象交互。中介者使各对象不需要显式地相互引用，从而使其耦合松散，而且可以独立地改变它们之间的交互。观察者（Observer）模式定义对象间的一种一对多的依赖关系，当一个对象的状态发生改变时，所有依赖于它的对象都得到通知并被自动更新。

以上四种设计模式都是行为设计模式。行为设计模式大多注重封装变化，当一个程序的某个方面的特征经常发生改变时，这些模式就定义一个封装这个方面的对象。这样，当该程序的其他部分依赖于这个方面时，它们都可以与此对象协作。这些模式通常定义一个抽象类来描述这些封装变化的对象，并且通常该模式依据这个对象来命名。例如：

一个 Strategy 对象封装一个算法，一个 Mediator 对象封装对象间的协议。Mediator 和 Observer 是相互竞争的模式，之间的差别是：Observer 通过引入 Observer 和 Subject 对象来分布通信，而 Mediator 对象则封装了其他对象间的通信。

参考答案

（43）C　　　（44）D

试题（45）

UML 图中，一张交互图显示一个交互，由一组对象及其之间的关系组成，包含它们之间可能传递的消息。<u>（45）</u>不是交互图。

（45）A．序列图　　B．对象图　　C．通信图　　D．时序图

试题（45）分析

本题考查统一建模语言（UML）的基本知识。

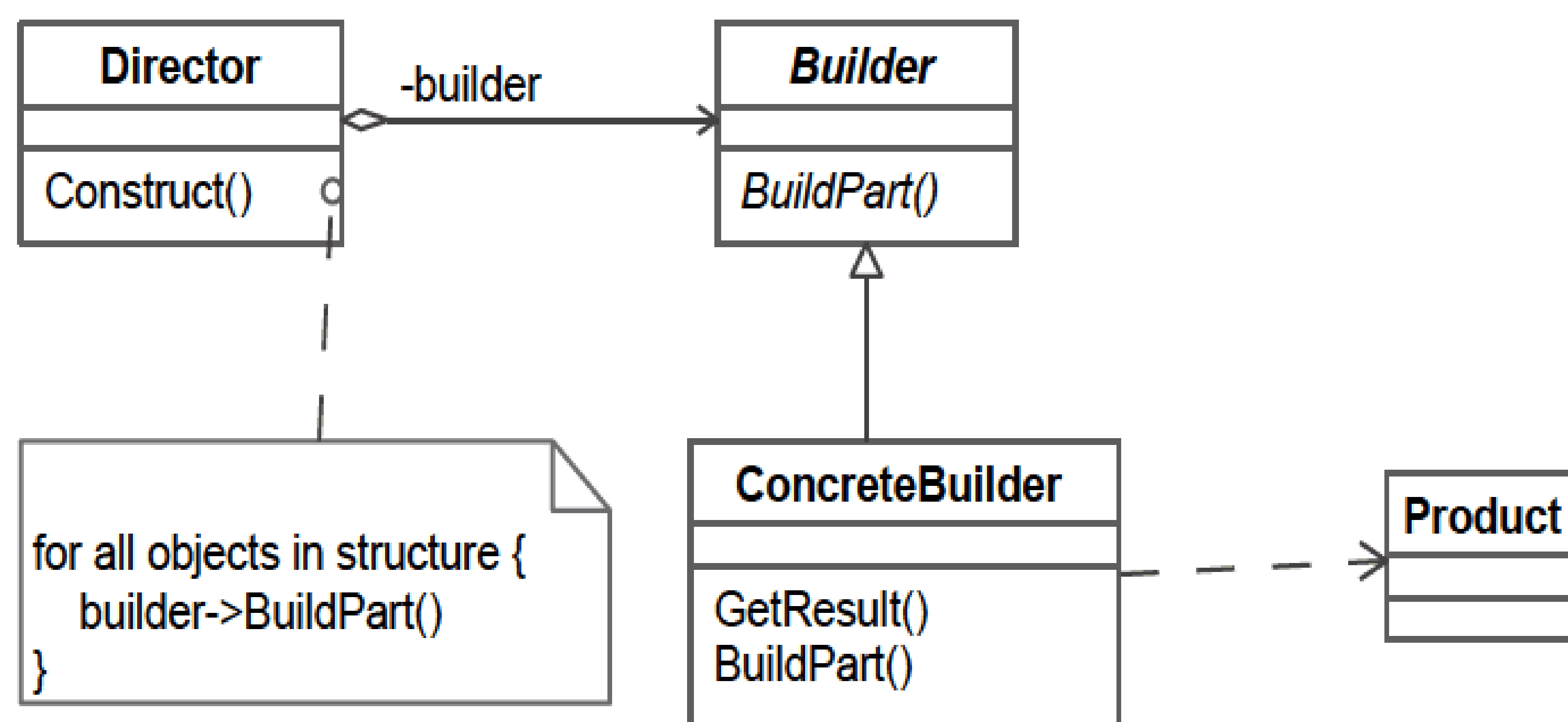
UML 中提供了多种建模系统的图，体现系统的静态方面和动态方面。对象图（object diagram）展现了某一时刻一组对象以及它们之间的关系。对象图描述了在类图中所建立的事物的实例的静态快照，给出系统的静态设计视图或静态进程视图。序列图（sequence diagram）是场景（scenario）的图形化表示，描述了以时间顺序组织的对象之间的交互活动。通信图（communication diagram）强调收发消息的对象的结构组织。时序图（Timing Diagram）关注沿着线性时间轴、生命线内部和生命线之间的条件改变，描述对象状态随着时间改变的情况，很像示波器，适合分析周期和非周期性任务。交互概览图（Interaction Overview Diagram）是 UML 2.0 新增的交互图之一，它是活动图的变体，描述业务过程中的控制流概览，软件过程中的详细逻辑概览，以及将多个图进行连接，抽象掉了消息和生命线。序列图、通信图、交互概览图和时序图均被称为交互图。

参考答案

（45）B

试题（46）、（47）

下图所示为<u>（46）</u>设计模式，适用于<u>（47）</u>。



（46）A．抽象工厂（Abstract Factory）　　B．生成器（Builder）

C．工厂方法（Factory Method）　　D．原型（Prototype）

（47）A．一个系统要由多个产品系列中的一个来配置时

B．当一个类希望由它的子类来指定它所创建的对象时

C．当创建复杂对象的算法应该独立于该对象的组成部分及其装配方式时

D．当一个系统应该独立于它的产品创建、构成和表示时

试题（46）、（47）分析

本题考查设计模式的基本概念。每种设计模式都有特定的意图，描述一个在我们周围不断重复发生的问题，以及该问题的解决方案的核心，使该方案能够重用而不必做重复劳动。

抽象工厂（Abstract Factory）模式提供一个创建一系列相关或相互依赖对象的接口，而无须指定他们具体的类。抽象工厂模式适用于一个系统要独立于它的产品的创建、组合和表示时；一个系统要由多个产品系列中的一个来配置时；当要强调一系列相关的产品对象的设计以便进行联合使用时；当提供一个产品类库，而只想显示它们的接口而不是实现时。

生成器（Builder）模式将一个复杂对象的构建与它的表示分离，使得同样的构建过程可以创建不同的表示。生成器模式适用于当创建复杂对象的算法应该独立于该对象的组成部分以及它们的装配方式时；当构造过程必须允许被构造的对象有不同的表示时。

工厂方法（Factory Method）定义一个用于创建对象的接口，让子类决定将哪一个类实例化，使一个类的实例化延迟到其子类。工厂方法模式适用于当一个类不知道它所必须创建的对象的类的时候；当一个类希望由它的子类来指定它所创建的对象的时候；当类将创建对象的职责委托给多个帮助子类中的某一个，并且你希望将哪一个帮助子类是代理者这一信息局部化的时候。

原型（Prototype）模式用原型实例指定创建对象的种类，并且通过拷贝这个原型来创建新的对象。原型模式适用于：当一个系统应该独立于它的产品创建、构成和表示时；当要实例化的类是在运行时刻指定时，例如通过动态装载，为了避免创建一个与产品类层次平行的工厂类层次时；当一个类的实例只能有几个不同状态组合中的一种时，建立相应数目的原型并克隆它们可能比每次用合适的状态手工实例化该类更方便一些。

参考答案

（46）B　　（47）C

试题（48）

对高级语言源程序进行编译的过程可以分为多个阶段，分配寄存器的工作在__（48）__阶段进行。

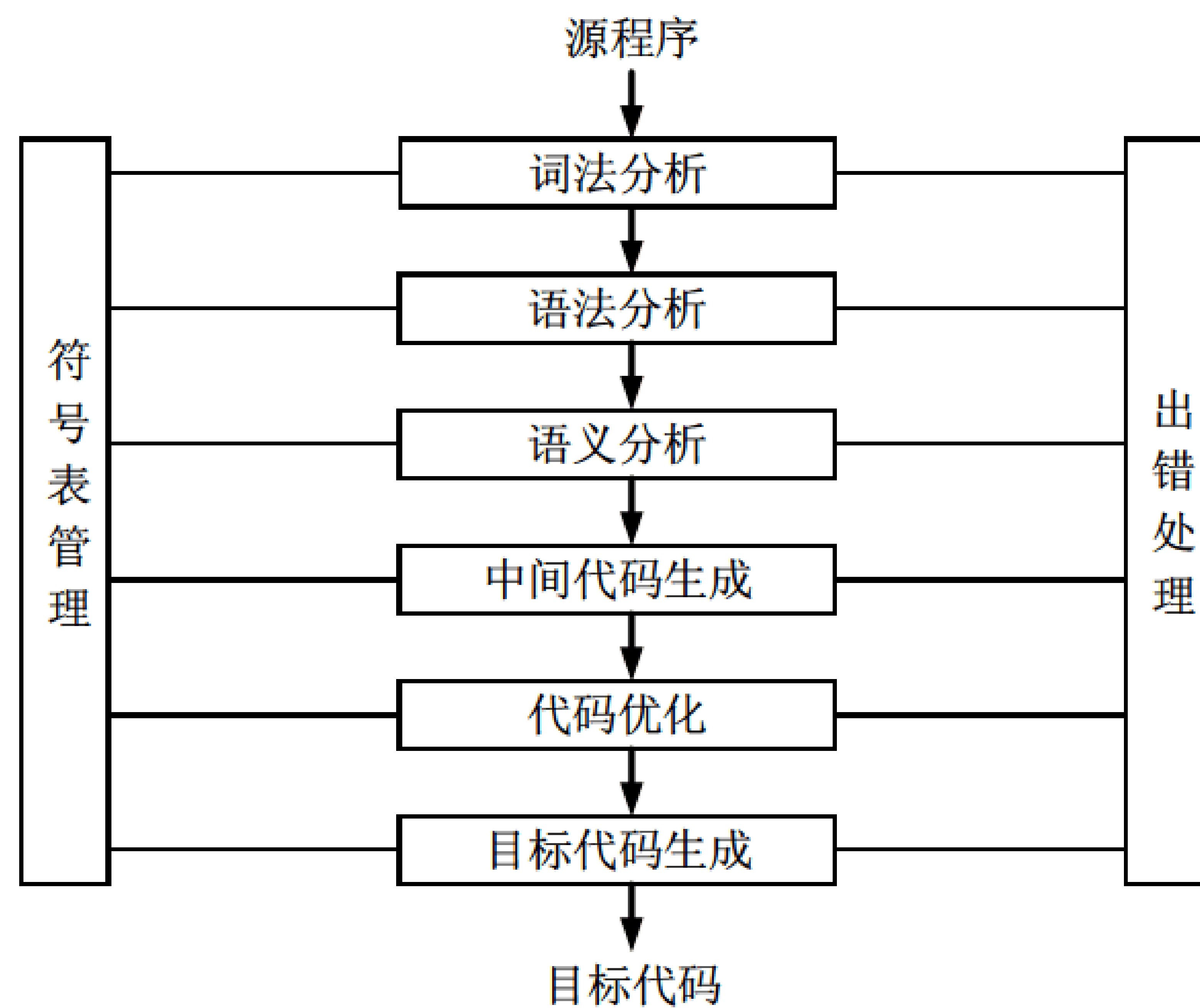
（48）A．词法分析　　B．语法分析

C．语义分析　　D．目标代码生成

试题（48）分析

本题考查程序语言基础知识。

编译程序的功能是把某高级语言书写的源程序翻译成与之等价的目标程序（汇编语言或机器语言）。编译程序的工作过程可以分为词法分析、语法分析、语义分析、中间代码生成、代码优化、目标代码生成、符号表管理和出错处理等部分，如下图所示。



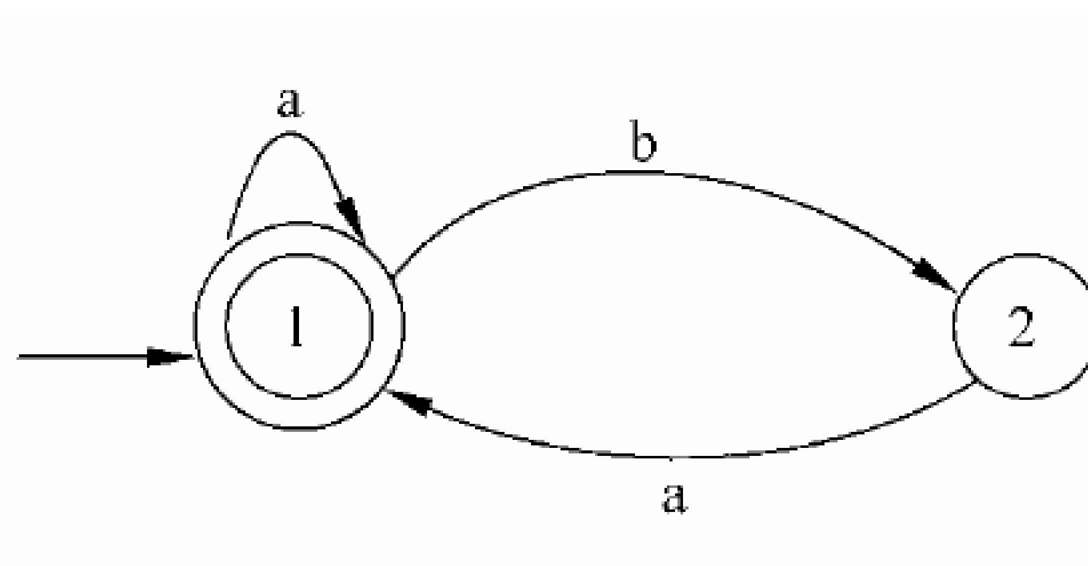
目标代码生成是编译器工作的最后一个阶段。这一阶段的任务是把中间代码变换成特定机器上的绝对指令代码、可重定位的指令代码或汇编指令代码，这个阶段的工作与具体的机器密切相关。因此在目标代码生成阶段分配寄存器。

参考答案

（48）D

试题（49）

以下关于下图所示有限自动机的叙述中，不正确的是___（49）___。



（49）A. 该自动机识别的字符串中 a 不能连续出现

B. 该自动机识别的字符串中 b 不能连续出现

C. 该自动机识别的非空字符串必须以 a 结尾

D. 该自动机识别的字符串可以为空串

试题（49）分析

本题考查程序语言基础知识。

自动机识别字符串的过程是：从初态出发，根据字符串的当前字符实现状态转移。如果存在从初态到终态的状态转移路径与字符串中的各个字符相匹配，那么就说该自动机可以识别该字符串。题中所给自动机的初态和终态都是编号为 1 的状态，从其状态图可知，从状态 1 开始，识别出字符“a”时仍然转移到状态 1，而识别出字符“b”时才离开状态 1 进入状态 2，状态 2 仅对字符“a”有状态转移，且转回状态 1。因此，该自动机识别的字符串仅包含 a、b 字符，但是字符“b”不能连续出现，连续出现“a”是可以的。

参考答案

（49）A

试题（50）

对于大多数通用程序设计语言，用＿（50）＿描述其语法即可。

（50）A. 正规文法　　B. 上下文无关文法
　　C. 上下文有关文法　　D. 短语结构文法

试题（50）分析

本题考查程序语言的基础知识。

乔姆斯基（Chomsky）把文法分成四种类型，即 0 型、1 型、2 型和 3 型。0 型文法也称为短语文法，其能力相当于图灵机，任何 0 型语言都是递归可枚举的；反之，递归可枚举集也必定是一个 0 型语言。1 型文法也称为上下文有关文法，这种文法意味着对非终结符的替换必须考虑上下文。2 型文法就是上下文无关文法，非终结符的替换无须考虑上下文。3 型文法等价于正规式，因此也被称为正规文法或线性文法。通用程序设计语言的大多数语法可由上下文无关文法表示。

参考答案

（50）B

试题（51）、（52）

在数据库逻辑结构设计阶段，需要＿（51）＿阶段形成的＿（52）＿作为设计依据。

（51）A. 需求分析　　B. 概念结构设计
　　C. 物理结构设计　　D. 数据库运行和维护

（52）A. 程序文档、数据字典和数据流图
　　B. 需求说明文档、程序文档和数据流图
　　C. 需求说明文档、数据字典和数据流图
　　D. 需求说明文档、数据字典和程序文档

试题（51）、（52）分析

本题考查数据库系统基本概念方面的基础知识。

数据库设计主要分为用户需求分析、概念结构、逻辑结构和物理结构设计四个阶段。其中，在用户需求分析阶段中，数据库设计人员采用一定的辅助工具对应用对象的功能、性能、限制等要求所进行的科学分析，并形成需求说明文档、数据字典和数据流程图。用户需求分析阶段形成的相关文档用以作为概念结构设计的设计依据。

参考答案

（51）A　　（52）C

试题（53）～（55）

给定关系模式 R(A, B, C, D)、S(C, D, E)，与 $\pi_{1,3,5}(\sigma_{2='软件工程'}(R \bowtie S))$ 等价的 SQL 语句如下：

```
SELECT  （53）  FROM R, S WHERE  （54） ;
```

（53）A．A, C, S.CB．A, B, E　　C．A, R.C, E　　D．A, R.C, S.D

（54）A．B＝ 软件工程 OR R.C =S.C AND R.D =S.D

B．B = '软件工程' OR R.C =S.C AND R.D =S.D

C．B = '软件工程' OR R.C =S.C OR R.D =S.D

D．B = '软件工程' AND R.C =S.C AND R.D =S.D

下列查询 B=“信息”且 E=“北京”的 A、B、E 的关系代数表达式中，查询效率最高的是 （55） 。

（55）A．$\pi_{1,2,7}(\sigma_{2='信息'\wedge 3=5\wedge 4=6\wedge 7='北京'}(R\times S))$

B．$\pi_{1,2,7}(\sigma_{3=5\wedge 4=6}(\sigma_{2='信息'}(R)\times\sigma_{5='北京'}(S)))$

C．$\pi_{1,2,7}(\sigma_{3=5\wedge 4=6\wedge 2='信息'}(R\times\sigma_{7='北京'}(S)))$

D．$\pi_{1,2,7}(\sigma_{3=5\wedge 4=6\wedge 7='北京'}(\sigma_{2='信息'}(R)\times S))$

试题（53）～（55）分析

本题考查关系代数运算与 SQL 查询方面的基础知识。

试题（53）的正确答案为选项 C。$\pi_{1,3,5}(\sigma_{2='软件工程'}(R \bowtie S))$ 的含义是从 $R \bowtie S$ 结果集中选取 B ='软件工程'的元组，再进行 R.A、R.C 和 S.E 投影。

试题（54）的正确答案为选项 D。自然联结 $R \bowtie S$ 中的公共属性为 C、D，所以在 SQL 中可以用条件“WHERE R.C =S.C AND R.D =S.D”来限定。对于选取运算 $\sigma_{2='信息化'}$ 在 SQL 中可以用条件“WHERE B ='软件工程'”来限定。

试题（55）的正确答案为选项 B。关系代数表达式查询优化的原则如下：

① 提早执行选取运算。对于有选择运算的表达式，应优化成尽可能先执行选择运算的等价表达式，以得到较小的中间结果，减少运算量和从外存读块的次数。

② 合并乘积与其后的选择运算为连接运算。在表达式中，当乘积运算后面是选择运算时，应该合并为连接运算，使选择与乘积一道完成，以避免做完乘积后，需再扫描一个大的乘积关系进行选择运算。

③ 将投影运算与其后的其他运算同时进行，以避免重复扫描关系。

④ 将投影运算和其前后的二目运算结合起来，使得没有必要为去掉某些字段再扫描一遍关系。

⑤ 在执行连接前对关系适当地预处理，就能快速地找到要连接的元组。方法有两种：索引连接法、排序合并连接法。

⑥ 存储公共子表达式。对于有公共子表达式的结果应存于外存（中间结果），这样，当从外存读出它的时间比计算的时间少时，就可节约操作时间。

显然，根据原则①尽量提早执行选取运算，正确的选项是 B。

参考答案

（53）C　（54）D　（55）B

试题（56）

给定关系模式 R(U, F)，U={ A, B, C, D, E, H }，函数依赖集 F={ A→B，A→C，C→D，AE→H}。关系模式 R 的候选关键字为＿（56）＿。

（56）A．AC　　B．AB　　C．AE　　D．DE

试题（56）分析

本题考查关系数据库基础知识。

试题（56）的正确答案为选项 C。关系模式 R 中，属性 AE 仅出现在函数依赖集 F 左部，而其余属性都不是左右都未出现的属性，所以 AE 必为 R 的唯一候选码。

参考答案

（56）C

试题（57）

对于线性表，相对于顺序存储，采用链表存储的缺点是＿（57）＿。

（57）A．数据元素之间的关系需要占用存储空间，导致存储密度不高

B．表中结点必须占用地址连续的存储单元，存储密度不高

C．插入新元素时需要遍历整个链表，运算的时间效率不高

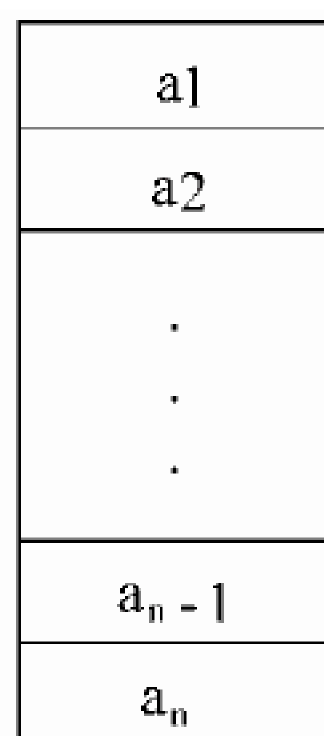
D．删除元素时需要遍历整个链表，运算的时间效率不高

试题（57）分析

本题考查数据结构基础知识。

对于线性表$(a_1,a_2,\cdots,a_n)$，顺序存储时表中元素占用的存储单元地址是连续的，因此逻辑上相邻的元素，其物理位置也相邻，如下图（a）所示。

线性表采用链式存储有单链表、双向链表、循环链表等形式，单链表如下图（b）所示。链式存储的基本特点是逻辑上相邻的元素不要求物理位置上相邻，所以需要在元素的存储单元中专门表示下一个（或上一个）元素的存储位置信息，从而可以得到元素间的顺序信息。





（a）线性表的顺序存储　　（b）线性表的单链表存储

参考答案

（57）A

试题（58）

若一个栈初始为空，其输入序列是 1, 2, 3, … , n-1, n，其输出序列的第一个元素为 k（$1 \leqslant k \leqslant \lfloor n/2 \rfloor$），则输出序列的最后一个元素是__（58）__。

（58）A．值为 n 的元素　　B．值为 1 的元素
C．值为 n-k 的元素　　D．不确定的

试题（58）分析

本题考查数据结构基础知识。

以 n 等于 4 举例说明。输入序列为 1 2 3 4，输出序列的第一个元素可以为 1 或 2。若为 1，则输出序列可能为 1 2 3 4、1 2 4 3、1 3 4 2、1 3 2 4、1 4 3 2；若为 2，则输出序列为 2 1 3 4、2 1 4 3、2 3 1 4、2 3 4 1、2 4 3 1。

以上序列都可由合法的入栈、出栈操作序列给出，从中可知无法确定输出序列中最后 1 个元素的值。

参考答案

（58）D

试题（59）

在某个二叉查找树（即二叉排序树）中进行查找时，效率最差的情形是该二叉查找树是__（59）__。

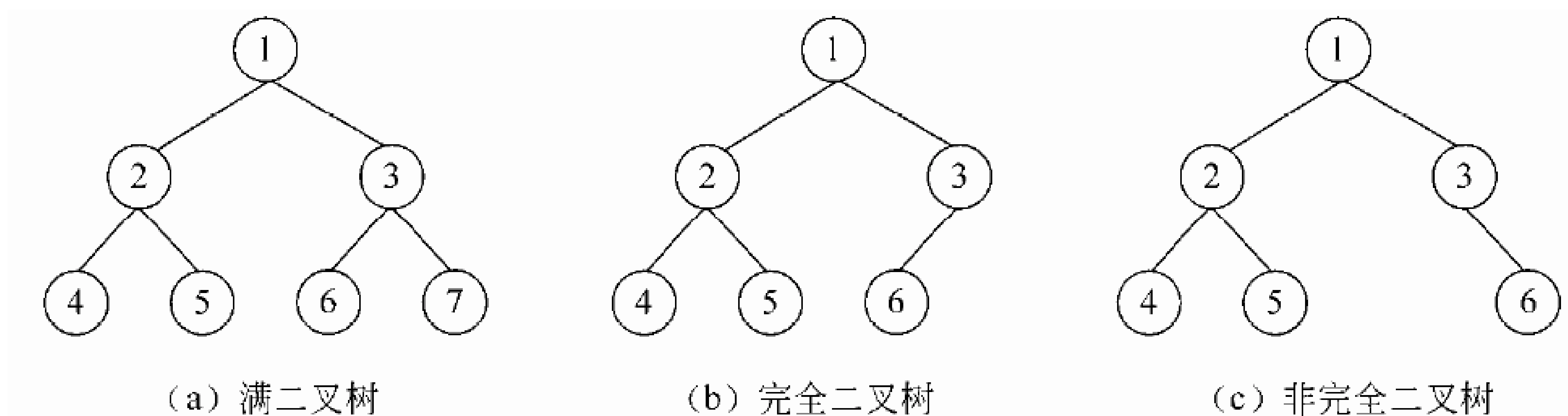
（59）A．完全二叉树　　B．平衡二叉树
C．单枝树　　D．满二叉树

试题（59）分析

本题考查数据结构基础知识。

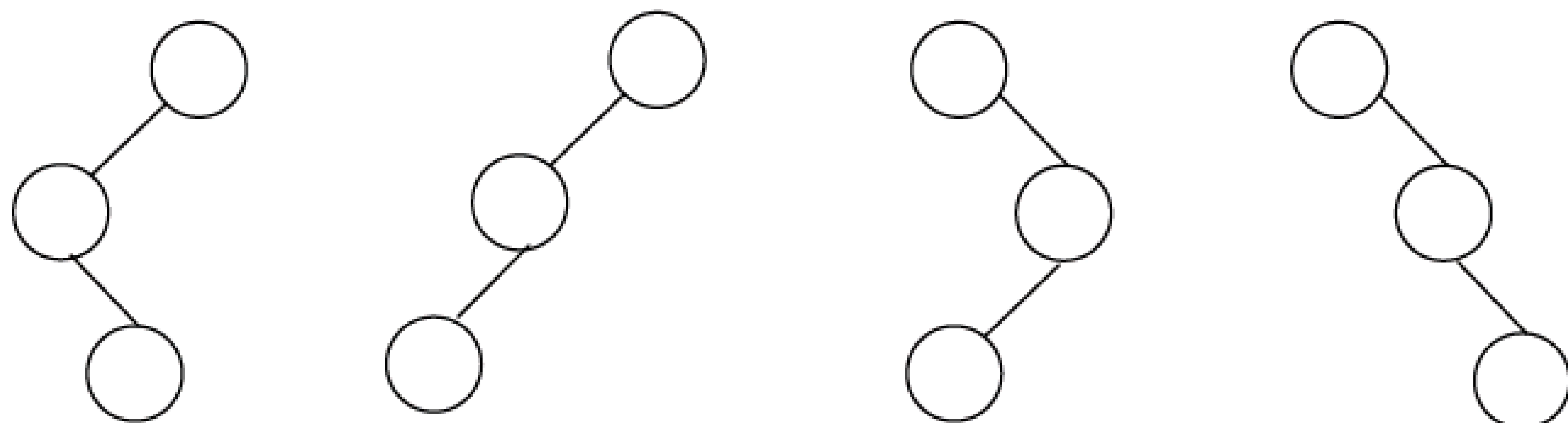
非空二叉查找树中的结点分布特点是左子树中的结点均小于树根，右子树中的结点均大于树根。因此，在二叉查找树中进行查找时，走了一条从树根出发到所找到结点的路径，到达一个空的子树则表明查找失败。

根据定义，高度为 h 的满二叉树中有 2^h-1 个结点，每一层上的结点数都达到最大值。完全二叉树的最高层只要求结点先占据左边的位置。例如，高度为 3 的满二叉树如下图（a）所示，具有 6 个结点的完全二叉树如下图（b）所示。



（a）满二叉树　（b）完全二叉树　（c）非完全二叉树

在平衡二叉树中，任何一个结点的左子树高度与右子树高度之差的绝对值不大于 1。单枝树中给每个结点只有 1 个子树。例如，具有 3 个结点的单枝树如下图所示。



显然，在结点数确定后，二叉查找树的形态为单枝树时查找效率最差。

参考答案

（59）C

试题（60）

在字符串的 KMP 模式匹配算法中，需先求解模式串的 next 函数值，其定义如下式所示，j 表示模式串中字符的序号（从 1 开始）。若模式串 p 为“abaac”，则其 next 函数值为　（60）　。

$$\text{next}[j]=\begin{cases}0 & j=1\\ \max\{k\,|\,1<k<j,'p_1p_2\cdots p_{k-1}'='p_{j-k+1}p_{j-k+2}\cdots p_{j-1}'\} & \\ 1 & \text{其他情况}\end{cases}$$

（60）A. 01234　　B. 01122　　C. 01211　　D. 01111

试题（60）分析

本题考查字符串的模式匹配运算知识。

KMP 是进行字符串模式匹配运算效率较高的算法。根据对 next 函数的定义，模式串前两个字符的 next 值为 0、1。对于第 3 个字符“a”，其在模式串中的前缀为“ab”，从该子串找不出前缀和后缀相同的部分，因此，根据定义，该位置字符的 next 值为 1。

对于第 4 个字符“a”，其在模式串中的前缀为“aba”，该子串只有长度为 1 的前缀

“a”和后缀“a”相同，根据定义，该位置字符的 next 值为 2。

对于第 5 个字符“a”，其在模式串中的前缀为“abaa”，该子串只有长度为 1 的前缀“a”和后缀“a”相同，根据定义，该位置字符的 next 值为 2。

综上可得，模式串“abaac”的 next 函数值为 01122。

参考答案

（60）B

试题（61）、（62）

快速排序算法在排序过程中，在待排序数组中确定一个元素为基准元素，根据基准元素把待排序数组划分成两个部分，前面一部分元素值小于等于基准元素，而后面一部分元素值大于基准元素。然后再分别对前后两个部分进一步进行划分。根据上述描述，快速排序算法采用了__（61）__算法设计策略。已知确定基准元素操作的时间复杂度为 $\Theta(n)$，则快速排序算法的最好和最坏情况下的时间复杂度为__（62）__。

（61）A．分治　B．动态规划　C．贪心　D．回溯

（62）A．$\Theta(n)$和$\Theta(nlgn)$　B．$\Theta(n)$和$\Theta(n^2)$

C．$\Theta(nlgn)$和$\Theta(nlgn)$　D．$\Theta(nlgn)$和$\Theta(n^2)$

试题（61）、（62）分析

本题考查快速排序算法。

快速排序算法是应用最为广泛的排序算法之一。其基本思想是将 n 个元素划分为两个部分：一部分元素值小于某个数；另一部分元素值大于某个数，该数的位置确定；然后进一步划分前面部分和后面部分。根据该叙述，可以知道，这里采用的是分治算法设计策略。

由于已知划分操作的时间复杂度为 $\Theta(n)$，不需要合并子问题的答案。对于最好的情况，应该是每次划分都把 n 个元素划分为大约 2 个 n/2 个元素的子数组，此时

$T(n) = 2T(n/2) + \Theta(n)$

解该递归式，可得时间复杂度为 $\Theta(nlgn)$。若刚好划分的极度不均衡，即每个划分刚好把 n 个元素划分为一边 0 个元素，一边 n-1 个元素，此时

$T(n) = T(n-1) + \Theta(n)$

解该递归式，可得时间复杂度为 $\Theta(n^2)$。

参考答案

（61）A　（62）D

试题（63）

对一待排序序列分别进行直接插入排序和简单选择排序，若待排序序列中有两个元素的值相同，则__（63）__保证这两个元素在排序前后的相对位置不变。

（63）A．直接插入排序和简单选择排序都可以

B．直接插入排序和简单选择排序都不能

C．只有直接插入排序可以

D．只有简单选择排序可以

试题（63）分析

本题考查简单排序算法特点。

直接插入排序的思想是：是将 n 个待排序的元素由一个有序表和一个无序表组成，开始时有序表中只包含一个元素。排序过程中，每次从无序表中取出第一个元素，将其插入到有序表中的适当位置，使有序表的长度不断加长，完成排序过程。

例如，对序列 21, 48, 21*, 9 进行直接插入排序，21 和 21*的相对位置在排序前后可保持，如下所示：

第一趟得到有序子序列：21, 48

第二趟得到有序子序列：21, 21*, 48

第三趟得到有序序列：9, 21,21*, 48

简单选择排序的过程是：第一趟在 n 个记录中选取最小记录作为有序序列的第一个记录；第二趟在 n-1 个记录中选取最小记录作为有序序列的第二个记录；第 i 趟在 n-i+1 个记录中选取最小的记录作为有序序列中的第 i 个记录，直到将序列排列有序。

对序列 21, 48, 21*, 9 进行简单选择排序，过程如下：

第一趟选出最小元素，将其交换至 1 号位置，序列为：**9**, 48, 21*, 21

第二趟选出次小元素，将其交换至 2 号位置，序列为：**9, 21***,48, 21

第三趟选出第三小元素，将其交换至 3 号位置，序列为：**9, 21*,21**, 48

从该例可知，简单选择排序过程不能保证排序码相同的两个元素在排序前后的相对位置不变，直接插入排序则可以。

参考答案

（63）C

试题（64）、（65）

已知一个文件中出现的各字符及其对应的频率如下表所示。若采用定长编码，则该文件中字符的码长应为__（64）__。若采用 Huffman 编码，则字符序列“face”的编码应为__（65）__。

字符	a	b	c	d	e	f
频率(%)	45	13	12	16	9	5

（64）A．2　　B．3　　C．4　　D．5

（65）A．110001001101　　B．001110110011

C．101000010100　　D．010111101011

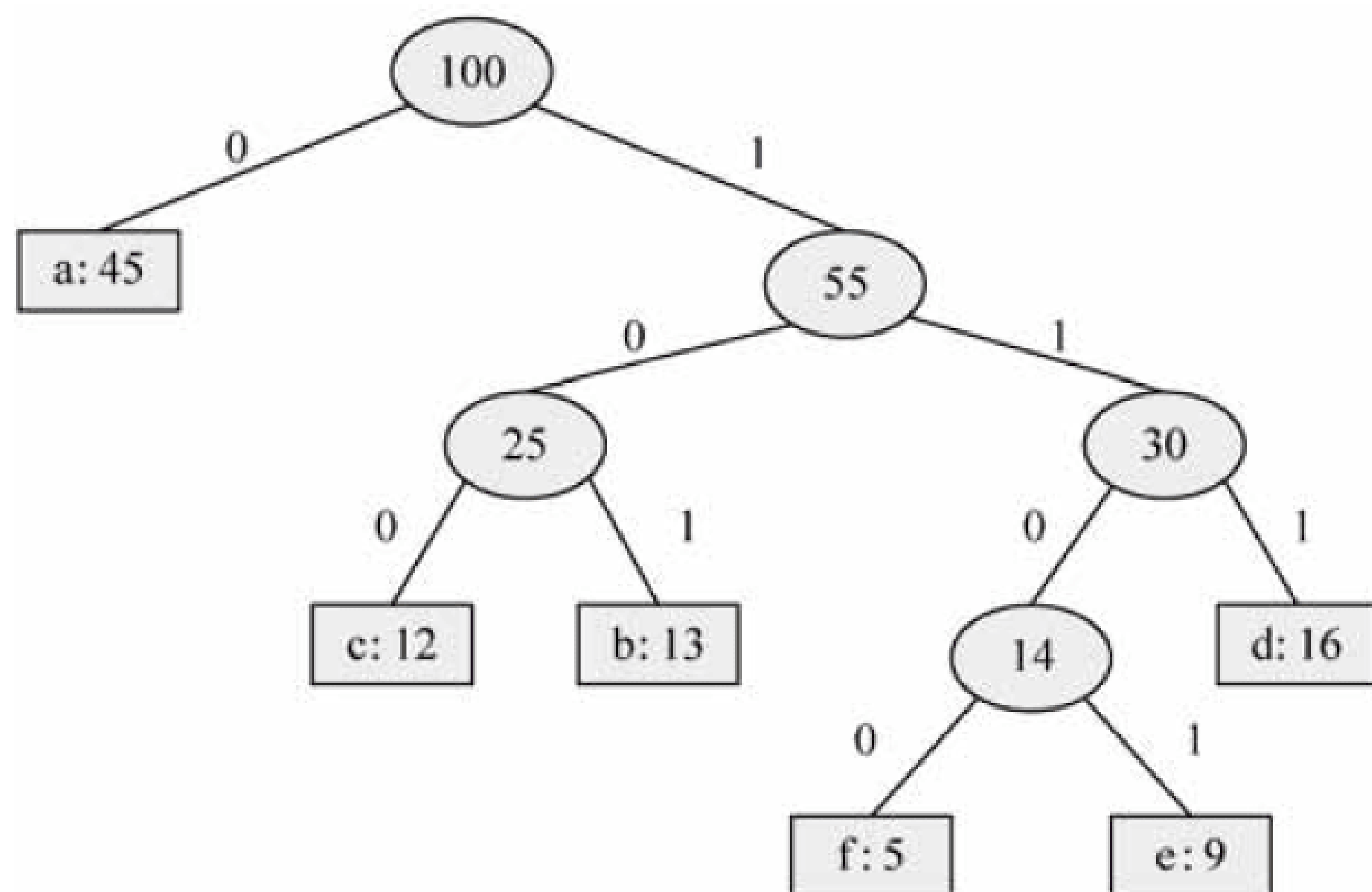
试题（64）、（65）分析

本题考查 Huffman 编码的相关知识。

字符在计算机中是用二进制表示的，每个字符用不同的二进制编码来表示。码的长度影响存储空间和传输效率。若是定长编码方法，用 2 位码长，只能表示 4 个字符，即

00、01、10 和 11；若用 3 位码长，则可以表示 8 个字符，即 000、001、010、011、100、101、110、111。对于题中给出的例子，一共有 6 个字符，因此采用 3 位码长的编码可以表示这些字符。

Huffman 编码是一种最优的不定长编码方法，可以有效的压缩数据。要使用 Huffman 编码，除了知道文件中出现的字符之外，还需要知道每个字符出现的频率。下图（a）是题干中给出对应的编码树，可以看到，每个字符及其对应编码为图（b），因此字符序列“face”的编码应为 1100 0 100 1101，即 65 选择 A。



（a）

字符	编码
a	0
b	101
c	100
d	111
e	1101
f	1100

（b）

参考答案

（64）B　　（65）A

试题（66）

PPP 中的安全认证协议是＿（66）＿，它使用三次握手的会话过程传送密文。

（66）A．MD5　　B．PAP　　C．CHAP　　D．HASH

试题（66）分析

PPP 认证是可选的。PPP 扩展认证协议（Extensible Authentication Protocol，EAP）可支持多种认证机制，并且允许使用后端服务器来实现复杂的认证过程。例如通过 Radius 服务器进行 Web 认证时，远程访问服务器（RAS）只是作为认证服务器的代理传递请求和应答报文，并且当识别出认证成功/失败标志后结束认证过程。通常 PPP 支持的两个认证协议是：

① 口令验证协议（Password Authentication Protocol，PAP）：提供了一种简单的两次握手认证方法，由终端发送用户标识和口令字，等待服务器的应答，如果认证不成功，则终止连接。这种方法不安全，因为采用文本方式发送密码，可能会被第三方窃取。

② 质询握手认证协议（Challenge Handshake Authentication Protocol，CHAP）：采用三次握手方式周期地验证对方的身份。首先是逻辑链路建立后认证服务器就要发送一个

挑战报文（随机数），终端计算该报文的 Hash 值并把结果返回服务器，然后认证服务器把收到的 Hash 值与自己计算的 Hash 值进行比较，如果匹配，则认证通过，连接得以建立，否则连接被终止。计算 Hash 值的过程有一个双方共享的密钥参与，而密钥是不通过网络传送的，所以 CHAP 是更安全的认证机制。在后续的通信过程中，每经过一个随机的间隔，这个认证过程都可能被重复，以缩短入侵者进行持续攻击的时间。值得注意的是，这种方法可以进行双向身份认证，终端也可以向服务器进行挑战，使得双方都能确认对方身份的合法性。

参考答案

（66）C

试题（67）、（68）

ICMP 协议属于因特网中的__（67）__协议，ICMP 协议数据单元封装在__（68）__中传送。

（67）A．数据链路层　B．网络层　C．传输层　D．会话层

（68）A．以太帧　B．TCP 段　C．UDP 数据报　D．IP 数据报

试题（67）、（68）分析

ICMP（Internet control Message Protocol）与 IP 协议同属于网络层，用于传送有关通信问题的消息，例如数据报不能到达目标站，路由器没有足够的缓存空间，或者路由器向发送主机提供最短通路信息等。ICMP 报文封装在 IP 数据报中传送，因而不保证可靠的提交。

参考答案

（67）B　（68）D

试题（69）

DHCP 客户端可从 DHCP 服务器获得__（69）__。

（69）A．DHCP 服务器的地址和 Web 服务器的地址

B．DNS 服务器的地址和 DHCP 服务器的地址

C．客户端地址和邮件服务器地址

D．默认网关的地址和邮件服务器地址

试题（69）分析

本试题考查 DHCP 协议的工作原理。

DHCP 客户端可从 DHCP 服务器获得本机 IP 地址、DNS 服务器的地址、DHCP 服务器的地址、默认网关的地址等，但没有 Web 服务器、邮件服务器地址。

参考答案

（69）B

试题（70）

分配给某公司网络的地址块是 210.115.192.0/20，该网络可以被划分为__（70）__个 C 类子网。

（70）A．4　　B．8　　C．16　　D．32

试题（70）分析

由于分配给公司网络的地址块是 210.115.192.0/20，留给子网掩码的比特数只有 4 位，所以只能划分为 16 个 C 类子网，这 16 个 C 类子网的子网号为 11000000～11001111，即 192～207，所以 210.115. 210. 0 不属于该公司的网络地址。

参考答案

（70）C

试题（71）～（75）

Teams are required for most engineering projects. Although some small hardware or software products can be developed by individuals, the scale and complexity of modern systems is such, and the demand for short schedules so great, that it is no longer __（71）__ for one person to do most engineering jobs. Systems development is a team __（72）__, and the effectiveness of the team largely determines the __（73）__ of the engineering.

Development teams often behave much like baseball or basketball teams. Even though they may have multiple specialties, all the members work toward __（74）__. However, on systems maintenance and enhancement teams, the engineers often work relatively independently, much like wrestling and track teams.

A team is __（75）__ just a group of people who happen to work together. Teamwork takes practice and it involves special skills. Teams require common processes; they need agreed-upon goals; and they need effective guidance and leadership. The methods for guiding and leading such teams are well known, but they are not obvious.

（71）A．convenient　　B．existing　　C．practical　　D．real

（72）A．activity　　B．job　　C．process　　D．application

（73）A．size　　B．quality　　C．scale　　D．complexity

（74）A．multiple objectives　　B．different objectives

　　C．a single objective　　D．independent objectives

（75）A．relatively　　B．/　　C．only　　D．more than

参考译文

大多数工程项目需要团队完成。虽然有些小规模的硬件或软件产品可以由个人完成，但是现代系统的规模大、复杂性高以及开发周期短的极高需求，使得一个人完成大多工程工作已经不再现实。系统开发是一个团队活动，团队的效率很大程度上决定工程

的质量。

开发团队经常表现的像是棒球队或篮球队，即使棒球队或篮球队可能有多种不同专长，但是所有的队员都朝着一个目标努力。然而在系统维护和增强团队，工程师们的工作就像摔跤和田径队一样经常相对独立。

团队不仅仅是一群人碰巧在一起工作，团队工作需要实践，涉及多种特殊的技能。团队需要共同的过程，需要达成一致的目标，需要有效地指导和领导。尽管指导和领导这样的团队的方法是众所周知的，但是它们并不明显。

参考答案

（71）C　（72）A　（73）B　（74）C　（75）D

第 24 章　2014 下半年软件设计师下午试题分析与解答

试题一（共 15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某大型匹萨加工和销售商为了有效管理生产和销售情况，欲开发一匹萨信息系统，其主要功能如下：

（1）销售。处理客户的订单信息，生成销售订单，并将其记录在销售订单表中。销售订单记录了订购者、所订购的匹萨、期望的交付日期等信息。

（2）生产控制。根据销售订单以及库存的匹萨数量，制定匹萨生产计划（包括生产哪些匹萨、生产顺序和生产量等），并将其保存在生产计划表中。

（3）生产。根据生产计划和配方表中的匹萨配方，向库存发出原材料申领单，将制作好的匹萨的信息存入库存表中，以便及时进行交付。

（4）采购。根据所需原材料及库存量，确定采购数量，向供应商发送采购订单，并将其记录在采购订单表中；得到供应商的供应量，将原材料数量记录在库存表中，在采购订单表中标记已完成采购的订单。

（5）运送。根据销售订单将匹萨交付给客户，并记录在交付记录表中。

（6）财务管理。在匹萨交付后，为客户开具费用清单，收款并出具收据；依据完成的采购订单给供应商支付原材料费用并出具支付细节；将收款和支付记录存入收支记录表中。

（7）存储。检查库存的原材料、匹萨和未完成订单，确定所需原材料。

现采用结构化方法对匹萨信息系统进行分析与设计，获得如图 1-1 所示的上下文数据流图和图 1-2 所示的 0 层数据流图。

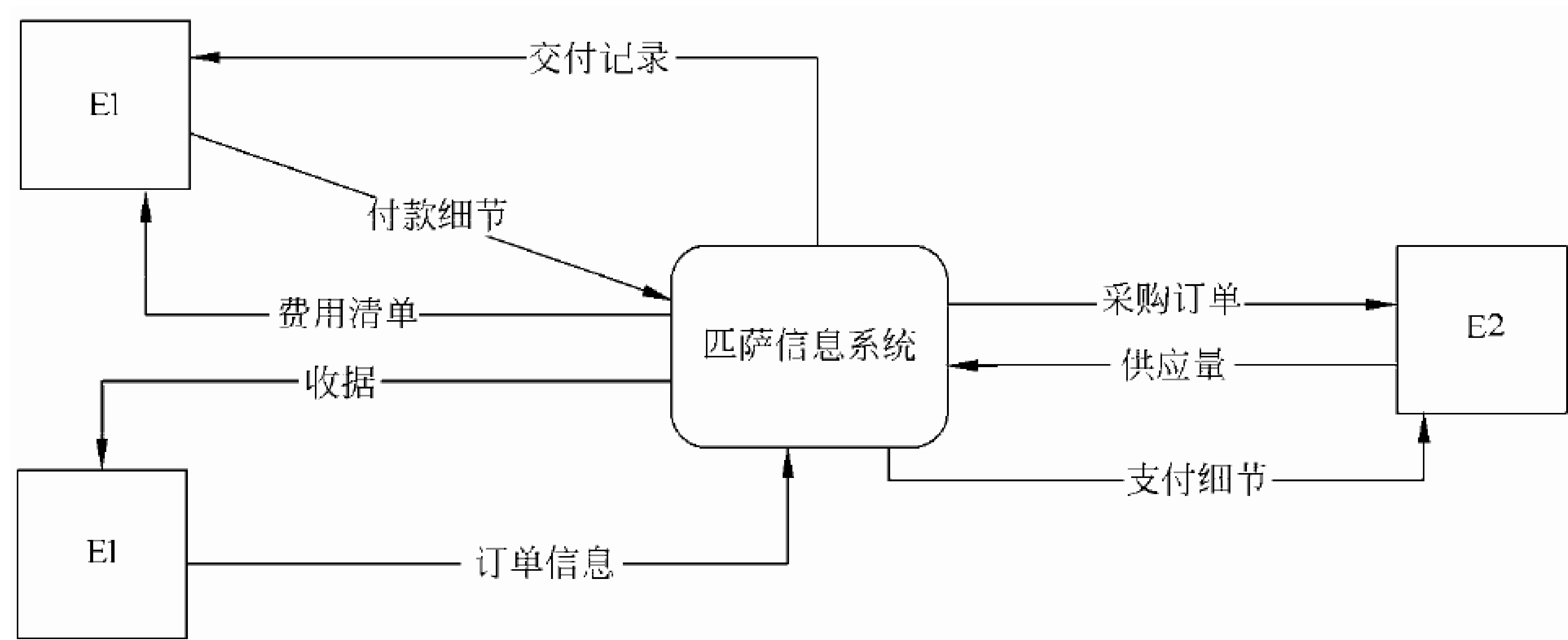


图 1-1　上下文数据流图

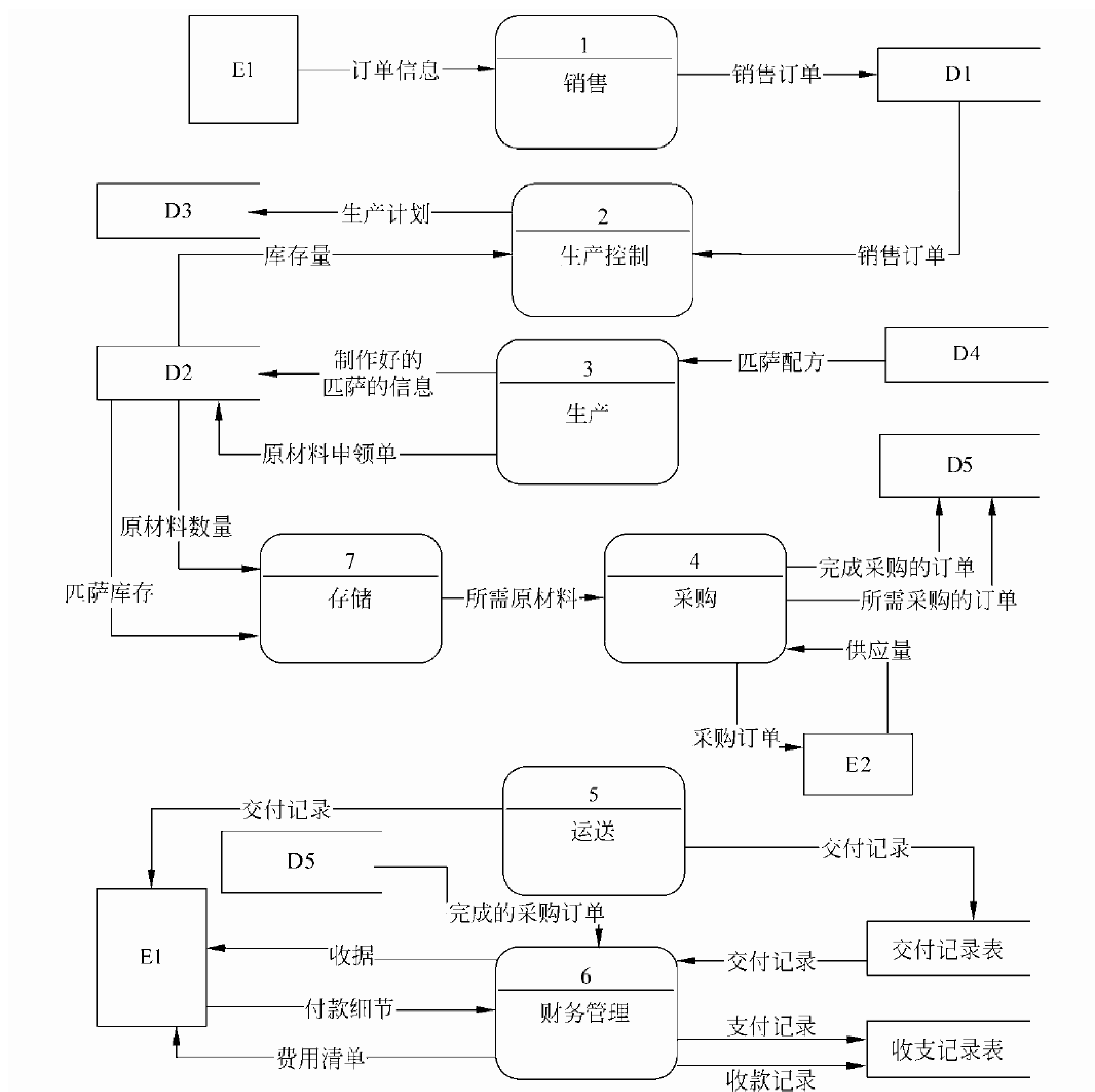


图 1-2　0 层数据流图

【问题 1】（4 分）

根据说明中的词语，给出图 1-1 中的实体 E1～E2 的名称。

【问题 2】（5 分）

根据说明中的词语，给出图 1-2 中的数据存储 D1～D5 的名称。

【问题 3】（6 分）

根据说明和图中词语，补充图 1-2 中缺失的数据流及其起点和终点。

试题一分析

本题考查采用结构化方法进行系统分析与设计，主要考查数据流图（DFD）的应用，

是比较传统的题目，考点与往年类似，要求考生细心分析题目中所描述的内容。

DFD 是一种便于用户理解、分析系统数据流程的图形化建模工具，是系统逻辑模型的重要组成部分。上下文 DFD（顶层 DFD）通常用来确定系统边界，将待开发系统看作一个大的加工（处理），然后根据系统从哪些外部实体接收数据流，以及系统将数据流发送到哪些外部实体，建模出的上下文图中只有唯一的一个加工和一些外部实体，以及这两者之间的输入输出数据流。0 层 DFD 在上下文确定的系统外部实体以及与外部实体的输入输出数据流的基础上，将上下文 DFD 中的加工分解成多个加工，识别这些加工的输入输出数据流，使得所有上下文 DFD 中的输入数据流，经过这些加工之后变换成上下文 DFD 的输出数据流。根据 0 层 DFD 中加工的复杂程度进一步建模加工的内容。

在建分层 DFD 时，根据需求情况可以将数据存储建模在不同层次的 DFD 中，注意要在绘制下层数据流图时要保持父图与子图平衡。父图中某加工的输入输出数据流必须与其子图的输入输出数据流在数量和名字上相同，或者父图中的一个输入（或输出）数据流对应于子图中几个输入（或输出）数据流，而子图中组成这些数据流的数据项全体正好是父图中的这一个数据流。

【问题 1】

本问题考查上下文 DFD，要求确定外部实体。考察系统的主要功能，不难发现，系统中涉及客户和供应商，没有提到其他与系统交互的外部实体。根据描述（1）中“处理客户的订单信息”等信息、从供应商出进行采购时“向供应商发送采购订单”，从而可确定 E1 为“客户”实体，E2 为“供应商”实体。

【问题 2】

本问题要求确定 0 层数据流图中的数据存储。分析说明中和数据存储有关的描述，根据说明中（1）中客户的订单信息，生成销售订单，“并将其记录在销售订单表”，可知 D1 为销售订单表；说明中（2）根据销售订单以及库存的匹萨数量制定匹萨生产计划，并将其保存在生产计划表中；说明（3）中 “根据生产计划和配方表中的匹萨配方”、“将制作好的匹萨的信息存入库存表中”，可知 D2 为库存表、D3 为生产计划表、D4 为匹萨配方表；说明（4）中向供应商发送采购订单，“并将其记录在采购订单表中”，可能 D5 为采购订单表。

【问题 3】

本问题要求补充缺失的数据流及其起点和终点。对照图 1-1 和图 1-2 的输入、输出数据流，数量不同，考查图 1-1 中输出至 E2 的数据流，有“采购订单”和“支付细节”，而图 1-2 中缺少了“支付细节”数据流，所以需要确定此数据流或者其分解的数据流的起点。

考查说明中的功能，功能（3）根据生产计划和配方表中的匹萨配方，向库存发出原材料申领单，对照图 1-2，不难发现，处理 3 缺少了输入流生产计划，而生产计划存储在“生产计划表”中。功能（4）根据所需原材料及库存量，确定采购数量，可以看出

处理 4 缺少了从库存表中获取的库存量；得到供应商的供应量，将原材料数量记录在库存表中，可以发现，缺少了从处理 4 到库存表的数据流“原材料数量”。图 1-2 中处理 5 没有输入流，考察功能（5）根据销售订单将匹萨交付给客户，可知，处理 5 的输入数据流为“销售订单”，其来源为数据存储订单记录表。功能（6）中依据完成的采购订单给供应商支付原材料费用并出具支付细节，可以看出缺少了支付细节输出流，结合对比图 1-1 和图 1-2，可知支付细节并未分解。功能（7）检查库存的原材料、匹萨和未完成订单，不难发现，处理 7 缺少输入流未完成订单，客户订单在数据存储“销售订单表”中。

试题一参考答案

【问题 1】

E1：客户　　　　　　E2：供应商

【问题 2】

D1：销售订单表　D2：库存表　D3：生产计划表　D4：配方表　D5：采购订单表

【问题 3】

数　据　流	起　　点	终　　点
生产计划	D3 或 生产计划表	3 或 生产
未完成订单	D1 或 销售订单表	7 或 存储
销售订单	D1 或 销售订单表	5 或 运送
原材料数量	4 或 采购	D2 或 库存表
库存量	D2 或 库存表	4 或 采购
支付细节	6 或 财务管理	E2 或 供应商

试题二（共 15 分）

阅读下列说明，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某集团公司在全国不同城市拥有多个大型超市，为了有效管理各个超市的业务工作，需要构建一个超市信息管理系统。

【需求分析结果】

（1）超市信息包括：超市名称、地址、经理和电话，其中超市名称唯一确定超市关系的每一个元组。每个超市只有一名经理。

（2）超市设有计划部、财务部、销售部等多个部门，每个部门只有一名部门经理，有多名员工，每个员工只属于一个部门。部门信息包括：超市名称、部门名称、部门经理和联系电话。超市名称、部门名称唯一确定部门关系的每一个元组。

（3）员工信息包括：员工号、姓名、超市名称、部门名称、职位、联系方式和工资。其中，职位信息包括：经理、部门经理、业务员等。员工号唯一确定员工关系的每一个元组。

（4）商品信息包括：商品号、商品名称、型号、单价和数量。商品号唯一确定商品关系的每一个元组。一名业务员可以负责超市内多种商品的配给，一种商品可以由多名业务员配给。

【概念模型设计】

根据需求分析阶段收集的信息，设计的实体联系图和关系模式（不完整）如下：

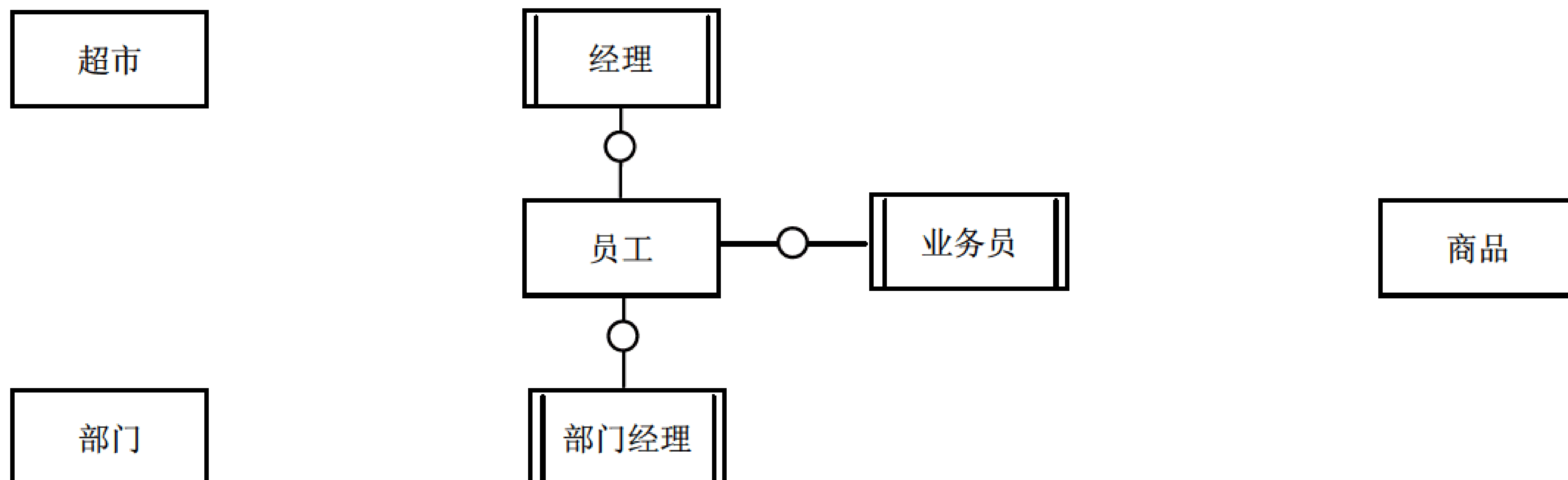


图 2-1　实体联系图

【关系模式设计】

超市（超市名称，经理，地址，电话）

部门（＿＿＿＿（a）＿＿＿＿，部门经理，联系电话）

员工（＿＿＿＿（b）＿＿＿＿，姓名，联系方式，职位，工资）

商品（商品号，商品名称，型号，单价，数量）

配给（＿＿＿＿（c）＿＿＿＿，配给时间，配给数量，业务员）

【问题 1】（4 分）

根据问题描述，补充四个联系，完善图 2-1 的实体联系图。联系名可用联系 1、联系 2、联系 3 和联系 4 代替，联系的类型分为 1:1、1:n 和 m:n（或 1:1、1:*和*:*）。

【问题 2】（7 分）

（1）根据实体联系图，将关系模式中的空（a）～（c）补充完整；

（2）给出部门和配给关系模式的主键和外键。

【问题 3】（4 分）

（1）超市关系的地址可以进一步分为邮编、省、市、街道，那么该属性是属于简单属性还是复合属性？请用 100 字以内文字说明。

（2）假设超市需要增设一个经理的职位，那么超市与经理之间的联系类型应修改为（d）＿＿，超市关系应修改为＿＿（e）＿＿。

试题二分析

本题考查数据库概念结构设计及其向逻辑结构转换的过程。

此类题目要求考生认真阅读题目对现实问题的描述，经过分类、聚集、概括等方法，

从中确定实体及其联系。题目已经给出了 4 个实体，需要根据需求描述，给出实体间的联系。

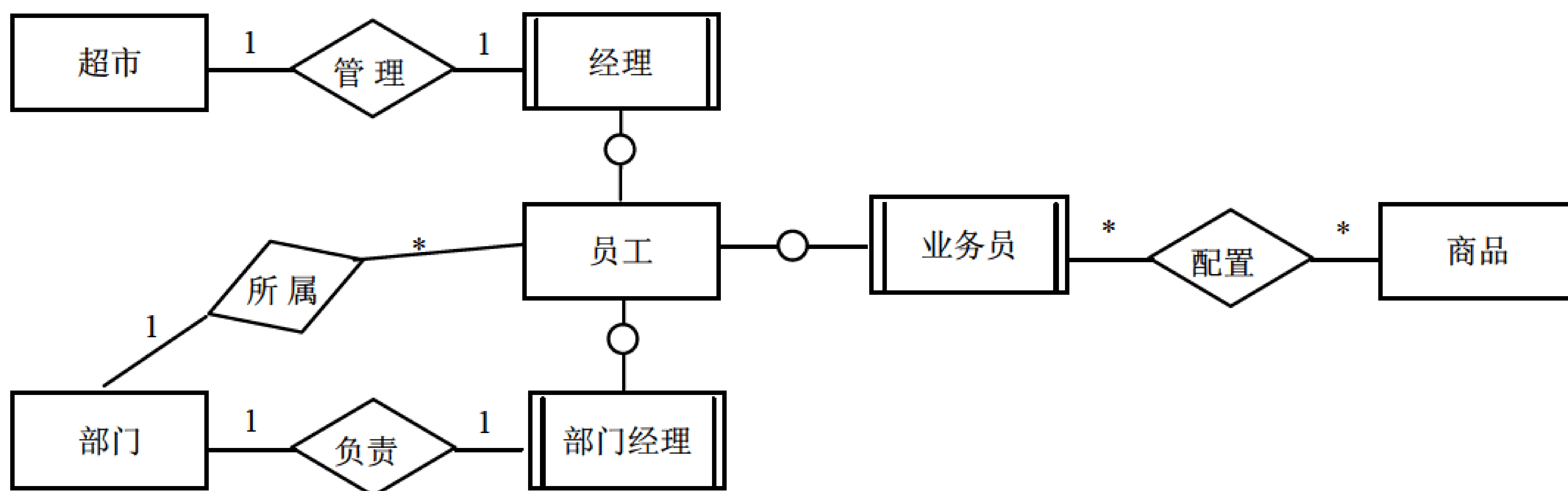
【问题 1】

根据题意，“每个超市有一位经理” 并且部门经理也是超市的员工，可以得出超市与经理之间的管理联系类型为 1:1。又由于“每个部门有一名部门经理”并且部门经理也是超市的员工，可以得出部门与部门经理之间的负责联系类型为 1:1。

由“每个部门有多名员工，每个员工属于一个部门”可以得出部门与员工间的所属联系类型为 1:*；并且员工是经理的超类型，经理是员工的子类型。

由“一名业务员可以负责超市内多种商品配置，一种商品可以由多名业务员配置”，可以得出，业务员与商品之间的配置联系类型为*:*。

完整的 ER 图如下：



【问题 2】

（1）完整的关系模式如下：

超市（超市名称，经理，地址，电话）

部门（超市名称，部门名称.，部门经理，联系电话）

员工（员工号，超市名称，部门名称，姓名，联系方式，职位，工资）

商品（商品号，商品名称，型号，单价，数量）

配置（商品号，配置时间，配置数量，业务员）

（2）部门和配置关系模式的主键和外键的分析如下：

在部门关系模式中，由于每个超市都设有计划部、财务部、销售部等多个部门，因此为了区分部门关系的每一个元组，需要“超市名称、部门名称”作为部门的主键。又因为部门经理也是员工，因此部门经理为员工关系的外键。

在配置关系模式中，“商品号，配置时间，业务员”唯一标识配置关系的每一个元组，故为配置关系的主键，外键为商品号，业务员。

【问题 3】

（1）超市的地址属性不属于简单属性。因为根据题意，超市关系的地址可以进一步

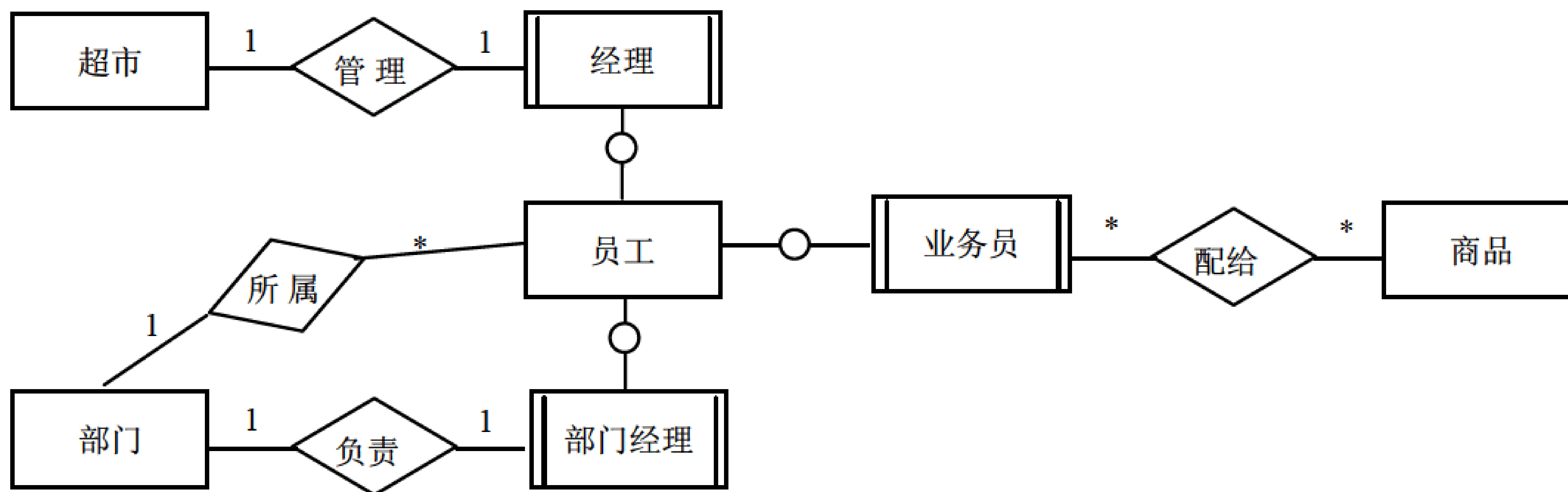
分为邮编、省、市、街道，而简单属性是原子的、不可再分的，复合属性可以细分为更小的部分（即划分为别的属性）。本小题的超市的地址可以进一步分为邮编、省、市、街道，故属于复合属性。

（2）因为根据题意，超市需要增设一位经理的职位，那么超市与经理之间的联系类型应修改为 1:n，超市的主键应修改为超市名称，经理，电话。

试题二参考答案

【问题 1】

联系名称可不做要求，但不能出现重名。



【问题 2】

（1）（a）超市名称，部门名称

（b）员工号，超市名称，部门名称

（c）商品号

（2）部门关系模式的主键：超市名称，部门名称

外键：部门经理

配给关系模式的主键：商品号，配给时间，业务员

外键：商品号，业务员

【问题 3】

（1）该属性属于复合属性，因为简单属性是原子的、不可再分的。

（2）（d）1:n

（e）超市名称，经理，电话

试题三（共 15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某公司欲开发一个管理选民信息的软件系统。系统的基本需求描述如下：

（1）每个人（Person）可以是一个合法选民（Eligible）或者无效的选民（Ineligible）。

（2）每个合法选民必须通过该系统对其投票所在区域（即选区，Riding）进行注册

（Registration）。每个合法选民仅能注册一个选区。

（3）选民所属选区由其居住地址（Address）决定。假设每个人只有一个地址，地址可以是镇（Town）或者城市（City）。

（4）某些选区可能包含多个镇，而某些较大的城市也可能包含多个选区。

现采用面向对象方法对该系统进行分析与设计，得到如图 3-1 所示的初始类图。

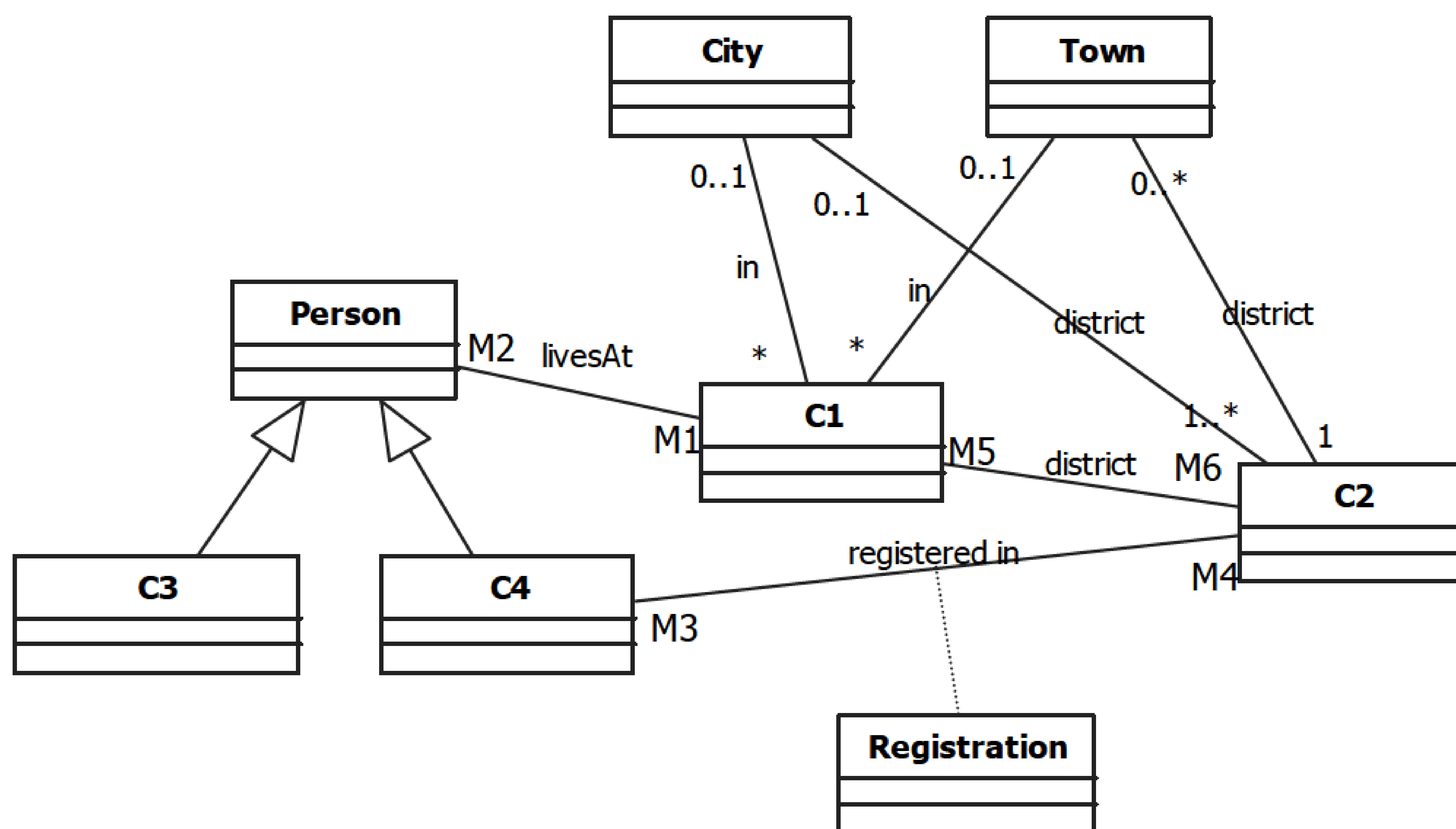


图 3-1　类图

【问题 1】（8 分）

根据说明中的描述，给出图 3-1 中 C1～C4 所对应的类名（类名使用说明中给出的英文词汇）。

【问题 2】（3 分）

根据说明中的描述，给出图 3-1 中 M1～M6 处的多重度。

【问题 3】（4 分）

现对该系统提出了以下新需求：

（1）某些人拥有在多个选区投票的权利，因此需要注册多个选区；

（2）对于满足（1）的选民，需要划定其“主要居住地”，以确定他们应该在哪个选区进行投票。

为了满足上述需求，需要对图 3-1 所示的类图进行哪些修改？请用 100 字以内文字说明。

试题三分析

本题属于经典的考题，主要考查面向对象分析方法与设计的基本概念与应用。在建

模方面，本题中只涉及到了 UML 类图。类图上的考点也是比较常规的对类的识别以及多重度的确定，题目难度不大。

【问题 1】

根据【说明】中的“（1）每个人（Person）可以是一个合法选民（Eligible）或者无效的选民（Ineligible）”，可以推断出这里有一个“一般/特殊”关系，应采用继承结构。再对照类图，C3、C4 处显而易见应该是 Ineligible 和 Eligible。由于 C4 和 C2 之间的关联关系，这里 C3 和 C4 的答案是不能互换的。

根据【说明】中的“（3）选民所属选区由其居住地址（Address）决定。假设每个人只有一个地址，地址可以是镇（Town）或者城市（City）”，可以推断出 C1、City、Town 这 3 个类描述的是与地址相关的内容，因此 C1 处应该是 Address。对应地，C2 处应该是 Riding，这个由 C2 与 City、C2 与 Town 之间的联系名称“district”也能推断出来。

【问题 2】

对于联系的多重度的判定，应注意题目中关于不同概念之间关联数量的描述。

M1、M2 这一对多重度，刻画的是“Person”和“Address”之间的关系。由【说明】中的“假设每个人只有一个地址”，可以得出 M1 和 M2 处分别为 1 和*。

M3 和 M4 描述的是合法选民与选区之间的关系。由【说明】中的“每个合法选民仅能注册一个选区”，可知 M3 和 M4 分别为*和 1。

M5 和 M6 描述的是选区和地址之间的关系。在【说明】中假设，每个合法选民在选区中只注册一个地址，因为 M5 和 M6 处分别为*和 1。

【问题 3】

本问题考查当需求发生变化时，对设计模型的修改。这里给出了两个需求变更，分别对初始类图进行修改。

需求 1：某些人拥有在多个选区投票的权利，因此需要注册多个选区。由于选区由住址确定，能够在多个选区注册，意味着其居住地址不止一个。所以“Person”和“Address”之间的多重度会发生变化。在选区注册时所使用的地址也不唯一了，因此需要增加属性来记录在注册选区时所使用的地址，从而对 C2 和 C4 之间的关联类进行修改，增加其属性。

需求 2：对于满足需求 1 的选民，需要划定其“主要居住地”，以确定他们应该在哪个选区进行投票。这个需求对选民的地址信息提出了更为详细的需求，按照面向对象方法将“不变部分和可变部分分离”的思想，在类图中增加一个新的类，并采用继承机制继承原有 Address 类中的共性元素。

试题三参考答案

【问题 1】

C1：Address　　C2：Riding　　C3：Ineligible　　C4：Eligible

【问题 2】

M1：1　M2：*或 0..*　M3：*　M4：1　M5：*　M6：1

【问题 3】

（1）M1 处改为 1..*，在 Registration 类中增加 address 属性，指明注册时使用的是哪个地址。

（2）增加一个类“主要居住地”，作为类 Address 的子类；类 Person 与类“主要居住地”之间具有关系联系，且每个人只有一个主要居住地。

试题四（共 15 分）

阅读下列说明和 C 代码，回答问题 1 至问题 3，将解答写在答题纸的对应栏内。

【说明】

计算一个整数数组 a 的最长递增子序列长度的方法描述如下：

假设数组 a 的长度为 n，用数组 b 的元素 b[i]记录以 a[i](0≤i<n)为结尾元素的最长递增子序列的长度，则数组 a 的最长递增子序列的长度为 $\max_{0\leqslant i<n}\{b[i]\}$；其中 b[i]满足最优子结构，可递归定义为：

$$\begin{cases} b[0]=1 \\ b[i]=\max\limits_{\substack{0\leqslant k\leqslant i \\ a[k]\leqslant a[i]}}\{b[k]\}+1 \end{cases}$$

【C 代码】

下面是算法的 C 语言实现。

（1）常量和变量说明

a：长度为 n 的整数数组，待求其最长递增子序列

b：长度为 n 的数组，b[i]记录以 a[i](0≤i<n)为结尾元素的最长递增子序列的长度，其中 0≤i<n

len：最长递增子序列的长度

i, j：循环变量

temp：临时变量

（2）C 程序

```
#include <stdio.h>
int maxL(int *b, int n){
    int i, temp = 0;
    for(i = 0; i < n; i++){
      if (b[i] > temp )
temp = b[i];
    }
    return temp;
```

```
    }

    int main(){
    int n, a[100], b[100], i, j, len;
    scanf("%d", &n);
        for(i = 0; i < n; i++){
           scanf("%d", &a[i]);
        }
        ____(1)____;
        for(i = 1; i < n; i++){
            for(j = 0,len = 0; ____(2)____; j++){
               if( ____(3)____ && len < b[j])
    len = b[j];
            }
            ____(4)____;
        }
        printf("len:%d\n", maxL(b,n));
        printf("\n");
    }
```

【问题 1】(8 分)

根据说明和 C 代码，填充 C 代码中的空（1）～（4）。

【问题 2】(4 分)

根据说明和 C 代码，算法采用了__（5）__设计策略，时间复杂度为__（6）__（用 O 符号表示）。

【问题 3】(3 分)

已知数组 a = {3, 10, 5, 15, 6, 8}，根据说明和 C 代码，给出数组 b 的元素值。

试题四分析

本题考查算法设计与分析以及用 C 程序设计语言来实现算法的能力。

此类题目要求考生认真阅读题目对问题的描述，理解算法思想，并会用 C 程序设计语言来实现。

【问题 1】

根据题干描述，用一个数组 b 来记录数组 a 每个子数组的最长递增子序列的长度，即 b[i]记录 a[0..i]的最长递增子序列的长度。首先，只有一个元素的数组的最长递增子序列的长度为 1，即给 b[0]直接赋值 1。因此，空（1）处填写“b[0] = 1”。两重 for 循环中，第一重是从 a 数组的第二个元素开始，考虑每个子数组 a[0..i]的最长递增子序列的长度，第二重是具体的计算过程。考虑子数组 a[0..i]，其最长递增子序列的长度应该等于子数组 a[0..i-1]中的比元素 a[i]小的元素的最长递增子序列的长度加 1，当然，可能存

在多个元素比元素 a[i]小，那么存在多个最长递增子序列的长度，此时，取最大者。因此，空处填写“j < i”，即考虑子数组 a[0..i-1]。空（3）处填写“a[j] <= a[i]”，要求元素值小于等于 a[i]而且目前的长度应该小于当前考虑的子数组的最长子序列长度。空（4）处填写“b[i] = len + 1”。简单地说，程序是根据题干给出的公式

$$\begin{cases} b[0]=1 \\ b[i]=\max\limits_{\substack{0\leqslant k\leqslant i \\ a[k]\leqslant a[i]}}\{b[k]\}+1 \end{cases}$$

来实现的。另外，计算的过程不是采用递归的方式，而是以一种自底向上的方式进行的。

【问题 2】

从题干说明和 C 程序来看，这是一个最优化问题，而且问题具有最优子结构，一个序列的最长递增子序列由其子序列的最长递增子序列构成。在计算过程中采用了自底向上的方式来进行，这具有典型的动态规划特征。因此采用的是动态规划设计策略。

C 程序中，有两重 for 循环，因此时间复杂度为 $O(n^2)$。

【问题 3】

输入数组为数组 a = {3, 10, 5, 15, 6, 8}，很容易得到，子数组 a[0..0]，a[0..1]，…,a[0..5]的最长递增子序列的长度分别为 1, 2, 2, 3, 3, 4，因此答案为 b = {1, 2, 2, 3, 3, 4}。该题可以根据题干说明、C 代码来计算。由于输入很简单，答案也可以从输入直接计算出来。

试题四参考答案

【问题 1】

（1）b[0] = 1

（2）j < i

（3）a[j] <= a[i]

（4）b[i] = len + 1

【问题 2】

（5）动态规划

（6）$O(n^2)$

【问题 3】

b = {1, 2, 2, 3, 3, 4}

试题五（共 15 分）

阅读下列说明和 C++代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

某灯具厂商欲生产一个灯具遥控器，该遥控器具有 7 个可编程的插槽，每个插槽都有开关按钮，对应着一个不同的灯。利用该遥控器能够统一控制房间中该厂商所有品牌

灯具的开关，现采用 Command（命令）模式实现该遥控器的软件部分。Command 模式的类图如图 5-1 所示。

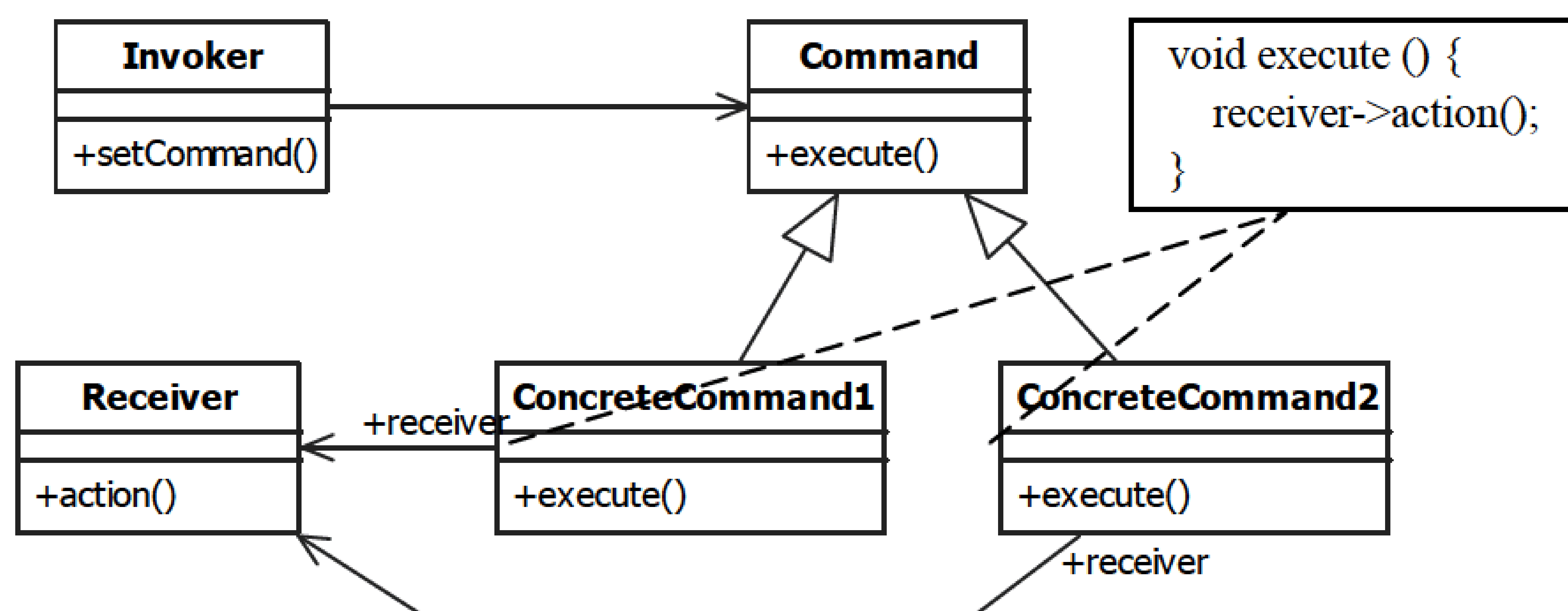


图 5-1　Command 模式类图

【C++代码】

```
class Light {
public:
  Light(string name) {  /* 代码省略 */  }
  void on()   {  /* 代码省略 */  }     //开灯
  void off()   {  /* 代码省略 */  }    //关灯
};
class Command {
public:
  ____(1)____;
};
class LightOnCommand : public Command {    //开灯命令
private:
   Light* light;
public:
   LightOnCommand(Light* light) {  this->light = light;   }
   void execute() {  ____(2)____;   }
};
class LightOffCommand : public Command {  //关灯命令
private:
  Light *light;
public:
  LightOffCommand(Light* light) {   this->light = light;  }
  void execute() {  ____(3)____;  }
};
```

```
class RemoteControl {   //遥控器
private:
   Command* onCommands[7];
   Command* offCommands[7];
public:
   RemoteControl()  {  /* 代码省略 */  }
   void setCommand(int slot, Command* onCommand, Command* offCommand) {
       ____(4)____ = onCommand;
       ____(5)____ = offCommand;
   }
   void onButtonWasPushed(int slot) { ____(6)____;   }
   void offButtonWasPushed(int slot) { ____(7)____;   }
};
int main() {
   RemoteControl* remoteControl = new RemoteControl();
Light* livingRoomLight = new Light("Living Room");
Light* kitchenLight = new Light("kitchen");
LightOnCommand* livingRoomLightOn = new LightOnCommand (livingRoomLight);
LightOffCommand* livingRoomLightOff = new LightOffCommand (livingRoomLight);
LightOnCommand* kitchenLightOn = new LightOnCommand(kitchenLight);
LightOffCommand* kitchenLightOff = new LightOffCommand(kitchenLight);
remoteControl->setCommand(0, livingRoomLightOn, livingRoomLightOff);
remoteControl->setCommand(1, kitchenLightOn, kitchenLightOff);
remoteControl->onButtonWasPushed(0);
remoteControl->offButtonWasPushed(0);
remoteControl->onButtonWasPushed(1);
remoteControl->offButtonWasPushed(1);
   /* 其余代码省略  */
 return 0;
}
```

试题五分析

本题考查命令（Command）模式的基本概念和应用。

命令模式把一个请求或者操作封装到一个对象中。命令模式允许系统使用不同的请求把客户端参数化，对请求排队或者记录请求日志，可以提供命令的撤销和恢复功能。

在软件系统中，行为请求者与行为实现者之间通常呈现一种紧耦合的关系。但在某些场合，比如要对行为进行记录撤销重做事务等处理，这种无法抵御变化的紧耦合是不合适的。这种情况下，使用 Command 模式将行为请求者与行为实现者进行解耦。

题目中给出了 Command 模式的类图，其中：

Command 类为所有命令声明了一个接口。调用命令对象的 execute()方法，就可以让接收者进行相关的动作。

ConcreteCommand 类定义了动作和接收者之间的绑定关系。调用者只要调用 execute()就可以发出请求，然后由 ConcreteCommand 调用接收者的一个或多个动作。

Invoker 持有一个命令对象，并在某个时间点调用命令对象的 execute()方法，将请求付诸实行。

Receiver 知道如何进行必要的工作，实现这个请求。任何类都可以当接收者。

了解了 Command 模式的内涵之后，下面来看程序。

由于 Command 类的主要作用是为所有的 ConcreteCommand 定义统一的接口，在 C++中通常采用抽象类来实现。C++的抽象类是至少具有一个纯虚拟函数的类。本题中的接口就是 execute()方法，所以（1）处要求填写的是纯虚拟函数 execute 的定义方式，即 virtual void execute() = 0。

类 LightOnCommand、LightOffCommand 对应的就是模式中的 ConcreteCommand。ConcreteCommand 中 execute()方法的代码在类图中已经给出，现在需要确定 receiver 是谁。类 Light 充当的是 Receiver，其中定义了两种 action：on 和 off。所以（2）、（3）对应代码分别为 light->on()、light->off()。

类 RemoteControl 对应的是模式中的 Invoker，在该类中设置需要控制的命令对象。（4）处对应的代码为 onCommands[slot]，设置“开灯”命令对象；（5）处对应的代码为 offCommands[slot]，设置“关灯”命令对象。类 RemoteControl 中的方法 onButtonWasPushed 和 offButtonWasPushed，分别完成对开灯、关灯命令对象的 execute 方法的调用，所以（6）、（7）处分别对应代码 onCommands[slot]->execute()、offCommands[slot]->execute()。

试题五参考答案

（1）virtual void execute() = 0

（2）light->on()

（3）light->off()

（4）onCommands[slot]

（5）offCommands[slot]

（6）onCommands[slot]->execute()

（7）offCommands[slot]->execute()

试题六（共 15 分）

阅读下列说明和 Java 代码，将应填入 __（n）__ 处的字句写在答题纸的对应栏内。

【说明】

某灯具厂商欲生产一个灯具遥控器，该遥控器具有 7 个可编程的插槽，每个插槽都有开关按钮，对应着一个不同的灯。利用该遥控器能够统一控制房间中该厂商所有品牌

灯具的开关，现采用 Command（命令）模式实现该遥控器的软件部分。Command 模式的类图如图 6-1 所示。

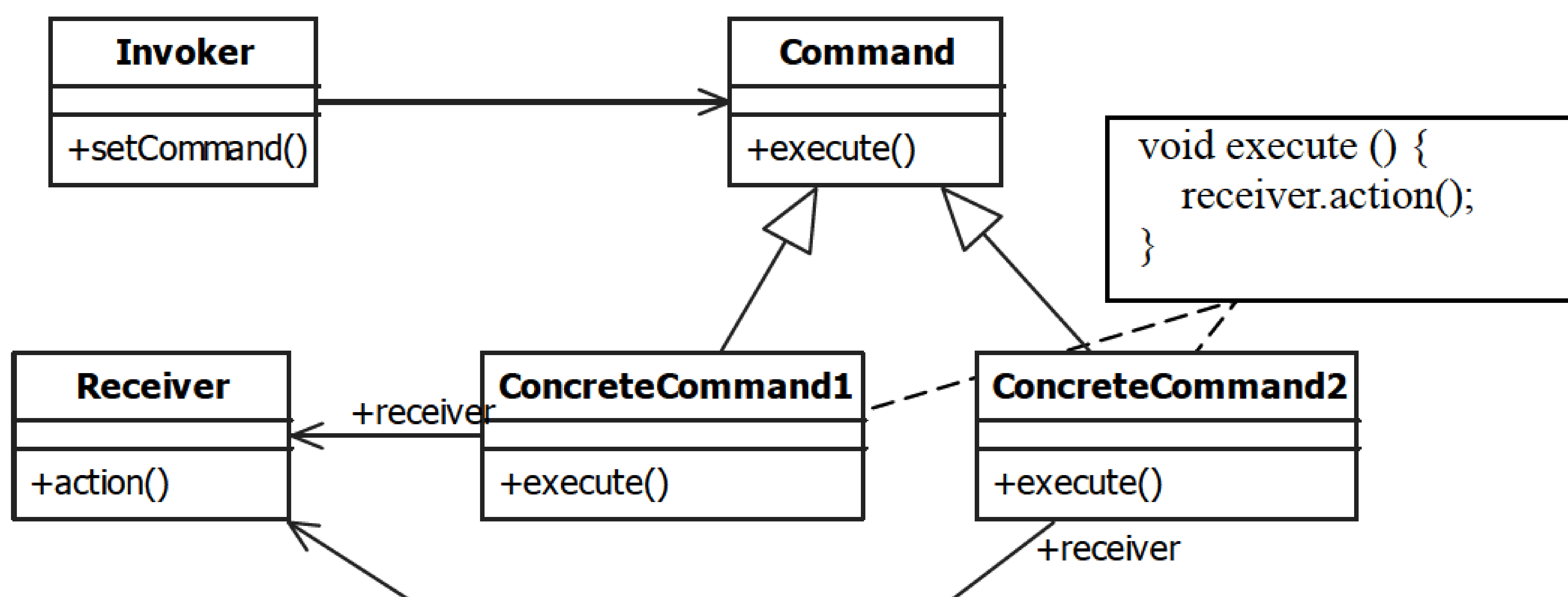


图 6-1　Command 模式类图

【Java 代码】

```
class Light {
  public Light() {}
  public Light(String name) {  /* 代码省略 */  }
  public void on()   {  /* 代码省略 */  }       //开灯
  public void off()   {  /* 代码省略 */  }      //关灯
  // 其余代码省略
}
_____(1)_____{
  public void execute();
}
class LightOnCommand implements Command {   //开灯命令
   Light light;
   public LightOnCommand(Light light) {  this.light = light;  }
   public void execute() {     _____(2)_____;   }
}
class LightOffCommand implements Command {  //关灯命令
  Light light;
  public LightOffCommand(Light light)  {   this.light = light;  }
  public void execute()  {     _____(3)_____;  }
}
class RemoteControl {                         //遥控器
   Command[] onCommands = new Command[7];
   Command[] offCommands = new Command[7];
```

```
    public RemoteControl(){  /* 代码省略 */  }
    public void setCommand(int slot, Command onCommand, Command offCommand) {
        ____(4)____ = onCommand;
        ____(5)____ = offCommand;
    }
    public void onButtonWasPushed(int slot) {
        ____(6)____;
    }
    public void offButtonWasPushed(int slot) {
        ____(7)____;
    }
}
class RemoteLoader {
  public static void main(String[] args) {
    RemoteControl remoteControl = new RemoteControl();
     Light livingRoomLight = new Light("Living Room");
     Light kitchenLight = new Light("kitchen");
     LightOnCommand livingRoomLightOn = new LightOnCommand(livingRoom
     Light);
     LightOffCommand livingRoomLightOff = new LightOffCommand(livingRoom
     Light);
     LightOnCommand kitchenLightOn = new LightOnCommand(kitchenLight);
     LightOffCommand kitchenLightOff=new LightOffCommand(kitchenLight);
     remoteControl.setCommand(0, livingRoomLightOn, livingRoomLightOff);
     remoteControl.setCommand(1, kitchenLightOn, kitchenLightOff);
     remoteControl.onButtonWasPushed(0);
     remoteControl.offButtonWasPushed(0);
     remoteControl.onButtonWasPushed(1);
     remoteControl.offButtonWasPushed(1);
  }
}
```

试题六分析

本题考查命令（Command）模式的基本概念和应用。

命令模式把一个请求或者操作封装到一个对象中。命令模式允许系统使用不同的请求把客户端参数化，对请求排队或者记录请求日志，可以提供命令的撤销和恢复功能。

在软件系统中，行为请求者与行为实现者之间通常呈现一种紧耦合的关系。但在某些场合，比如要对行为进行记录撤销重做事务等处理，这种无法抵御变化的紧耦合是不合适的。这种情况下，使用 Command 模式将行为请求者与行为实现者进行解耦。

题目中给出了 Command 模式的类图，其中：

Command 类为所有命令声明了一个接口。调用命令对象的 execute()方法，就可以让接收者进行相关的动作。

ConcreteCommand 类定义了动作和接收者之间的绑定关系。调用者只要调用 execute()就可以发出请求，然后由 ConcreteCommand 调用接收者的一个或多个动作。

Invoker 持有一个命令对象，并在某个时间点调用命令对象的 execute()方法，将请求付诸实行。

Receiver 知道如何进行必要的工作，实现这个请求。任何类都可以当接收者。

了解了 Command 模式的内涵之后，下面来看程序。

由于 Command 类的主要作用是为所有的 ConcreteCommand 定义统一的接口，在 Java 中通常采用接口（Interface）来实现，所以（1）处对应的代码为 interface Command。

类 LightOnCommand、LightOffCommand 对应的就是模式中的 ConcreteCommand。ConcreteCommand 中 execute()方法的代码在类图中已经给出，现在需要确定 receiver 是谁。类 Light 充当的是 Receiver，其中定义了两种 action：on 和 off。所以（2）、（3）对应代码分别为 light.on()和 light.off()。

类 RemoteControl 对应的是模式中的 Invoker，在该类中设置需要控制的命令对象。（4）处对应的代码为 onCommands[slot]，设置“开灯”命令对象；（5）处对应的代码为 offCommands[slot]，设置“关灯”命令对象。类 RemoteControl 中的方法 onButtonWasPushed 和 offButtonWasPushed，分别完成对开灯、关灯命令对象的 execute 方法的调用。所以（6）、（7）处分别对应代码 onCommands[slot].execute()、offCommands[slot].execute()。

试题六参考答案

（1）interface Command

（2）light.on()

（3）light.off()

（4）onCommands[slot]

（5）offCommands[slot]

（6）onCommands[slot].execute()

（7）offCommands[slot].execute()

第 25 章　2015 上半年软件设计师上午试题分析与解答

试题（1）

机器字长为 n 位的二进制数可以用补码来表示＿（1）＿个不同的有符号定点小数。

（1）A．2^n　　B．2^{n-1}　　C．2^n-1　　D．$2^{n-1}+1$

试题（1）分析

本题考查计算机系统的基础知识。

二进制数据在计算机系统中的表示方法是最基本的专业知识。补码本身是带符号位的，补码表示的数字中 0 是唯一的，不像原码有 +0 和–0 之分，也就意味着 n 位二进制编码可以表示 2^n 个不同的数。

参考答案

（1）A

试题（2）

计算机中 CPU 对其访问速度最快的是＿（2）＿。

（2）A．内存　　B．Cache　　C．通用寄存器　　D．硬盘

试题（2）分析

本题考查计算机系统的基础知识。

计算机系统中的 CPU 内部对通用寄存器的存取操作是速度最快的，其次是 Cache，内存的存取速度再次，选项中访问速度最慢的就是作为外存的硬盘。它们共同组成分级存储体系来解决存储容量、成本和速度之间的矛盾。

参考答案

（2）C

试题（3）

Cache 的地址映像方式中，发生块冲突次数最小的是＿（3）＿。

（3）A．全相联映像　　B．组相联映像　　C．直接映像　　D．无法确定的

试题（3）分析

本题考查计算机组成的基础知识。

Cache 工作时，需要拷贝主存信息到 Cache 中，就需要建立主存地址和 Cache 地址的映射关系。Cache 的地址映射方法主要有三种，即全相联映像、直接映像和组相联映像。其中全相联方式意味着主存的任意一块可以映像到 Cache 中的任意一块，其特点是块冲突概率低，Cache 空间利用率高，但是相联目录表容量大导致成本高、查表速度慢；直接映像方式是指主存的每一块只能映像到 Cache 的一个特定的块中，整个 Cache 地址

与主存地址的低位部分完全相同，其特点是硬件简单，不需要相联存储器，访问速度快（无须地址变换），但是 Cache 块冲突概率高导致 Cache 空间利用率很低；组相联方式是对上述两种方式的折中处理，对 Cache 分组，实现组间直接映射，组内全相联，从而获得较低的块冲突概率、较高的块利用率，同时得到较快的速度和较低的成本。

参考答案

（3）A

试题（4）

计算机中 CPU 的中断响应时间指的是__（4）__的时间。

（4）A．从发出中断请求到中断处理结束
B．从中断处理开始到中断处理结束
C．CPU 分析判断中断请求
D．从发出中断请求到开始进入中断处理程序

试题（4）分析

本题考查计算机系统的基础知识。

中断系统是计算机实现中断功能的软硬件总称。一般在 CPU 中设置中断机构，在外设接口中设置中断控制器，在软件上设置相应的中断服务程序。中断源在需要得到 CPU 服务时，请求 CPU 暂停现行工作转向为中断源服务，服务完成后，再让 CPU 回到原工作状态继续完成被打断的工作。中断的发生起始于中断源发出中断请求，中断处理过程中，中断系统需要解决一系列问题，包括中断响应的条件和时机，断点信息的保护与恢复，中断服务程序入口、中断处理等。中断响应时间，是指从发出中断请求到开始进入中断服务程序所需的时间。

参考答案

（4）D

试题（5）

总线宽度为 32bit，时钟频率为 200MHz，若总线上每 5 个时钟周期传送一个 32bit 的字，则该总线的带宽为__（5）__MB/s。

（5）A．40　　B．80　　C．160　　D．200

试题（5）分析

本题考查计算机系统的基础知识。

总线宽度是指总线的线数，即数据信号的并行传输能力，也体现总线占用的物理空间和成本；总线的带宽是指总线的最大数据传输率，即每秒传输的数据总量。总线宽度与时钟频率共同决定了总线的带宽。

$$32\text{bit} / 8 = 4\ \text{Byte},\ 200\text{MHz}/5 \times 4\ \text{Byte} = 160\ \text{MB/s}$$

参考答案

（5）C

试题（6）

以下关于指令流水线性能度量的叙述中，错误的是__（6）__。

（6）A. 最大吞吐率取决于流水线中最慢一段所需的时间

B. 如果流水线出现断流，加速比会明显下降

C. 要使加速比和效率最大化应该对流水线各级采用相同的运行时间

D. 流水线采用异步控制会明显提高其性能

试题（6）分析

本题考查计算机系统结构的基础知识。

对指令流水线性能的度量主要有吞吐率，加速比和效率等指标。吞吐率是指单位时间内流水线所完成的任务数或输出结果的数量，最大吞吐率则是流水线在达到稳定状态后所得到的吞吐率，它取决于流水线中最慢一段所需的时间，所以该段成为流水线的瓶颈。流水线的加速比定义为等功能的非流水线执行时间与流水线执行时间之比，加速比与吞吐率成正比，如果流水线断流，实际吞吐率将会明显下降，则加速比也会明显下降。流水线的效率是指流水线的设备利用率，从时空图上看效率就是 n 个任务所占的时空区与 m 个段总的时空区之比。因此要使加速比和效率最大化应该对流水线各级采用相同的运行时间。另外，流水线采用异步控制并不会给流水线性能带来改善，反而会增加控制电路的复杂性。

参考答案

（6）D

试题（7）

__（7）__协议在终端设备与远程站点之间建立安全连接。

（7）A. ARP　　B. Telnet　　C. SSH　　D. WEP

试题（7）分析

终端设备与远程站点之间建立安全连接的协议是 SSH。SSH 为 Secure Shell 的缩写，是由 IETF 制定的建立在应用层和传输层基础上的安全协议。SSH 是专为远程登录会话和其他网络服务提供安全性的协议。利用 SSH 协议可以有效防止远程管理过程中的信息泄露问题。SSH 最初是 UNIX 上的程序，后来又迅速扩展到其他操作平台。

参考答案

（7）C

试题（8）、（9）

安全需求可划分为物理线路安全、网络安全、系统安全和应用安全。下面的安全需求中属于系统安全的是__（8）__，属于应用安全的是__（9）__。

（8）A. 机房安全　　B. 入侵检测

C. 漏洞补丁管理　　D. 数据库安全

（9）A．机房安全　　B．入侵检测
C．漏洞补丁管理　　D．数据库安全

试题（8）、（9）分析

机房安全属于物理安全，入侵检测属于网络安全，漏洞补丁管理属于系统安全，而数据库安全则是应用安全。

参考答案

（8）C　（9）D

试题（10）

王某是某公司的软件设计师，每当软件开发完成后均按公司规定编写软件文档，并提交公司存档。那么该软件文档的著作权__（10）__享有。

（10）A．应由公司
B．应由公司和王某共同
C．应由王某
D．除署名权以外，著作权的其他权利由王某

试题（10）分析

本题考查知识产权的基本知识。

依据著作权法第十一条、第十六条规定，职工为完成所在单位的工作任务而创作的作品属于职务作品。职务作品的著作权归属分为两种情况。

① 虽是为完成工作任务而为，但非经法人或其他组织主持，不代表其意志创作，也不由其承担责任的职务作品，如教师编写的教材，著作权应由作者享有，但法人或者其他组织在其业务范围内有优先使用的权利，期限为 2 年。

② 由法人或者其他组织主持，代表法人或者其他组织意志创作，并由法人或者其他组织承担责任的职务作品，如工程设计、产品设计图纸及其说明、计算机软件、地图等职务作品，以及法律规定或合同约定著作权由法人或非法人单位单独享有的职务作品，作者享有署名权，其他权利由法人或者其他组织享有。

参考答案

（10）A

试题（11）

甲、乙两公司的软件设计师分别完成了相同的计算机程序发明，甲公司先于乙公司完成，乙公司先于甲公司使用。甲、乙公司于同一天向专利局申请发明专利。此情形下，__（11）__可获得专利权。

（11）A．甲公司　　B．甲、乙公司均
C．乙公司　　D．由甲、乙公司协商确定谁

试题（11）分析

本题考查知识产权的基本知识。

当两个以上的申请人分别就同样的发明创造申请专利的，专利权授给最先申请的

人。如果两个以上申请人在同一日分别就同样的发明创造申请专利的，应当在收到专利行政管理部门的通知后自行协商确定申请人。如果协商不成，专利局将驳回所有申请人的申请，即均不授予专利权。我国专利法规定："两个以上的申请人分别就同样的发明创造申请专利的，专利权授予最先申请的人。"我国专利法实施细则规定："同样的发明创造只能被授予一项专利。依照专利法第九条的规定，两个以上的申请人在同一日分别就同样的发明创造申请专利的，应当在收到国务院专利行政部门的通知后自行协商确定申请人。"

参考答案

（11）A

试题（12）

以下媒体中，（12）是感觉媒体。

（12）A. 音箱　B. 声音编码　C. 电缆　D. 声音

试题（12）分析

本题考查多媒体的基本知识。

感觉媒体指直接作用于人的感觉器官，使人产生直接感觉的媒体，如引起听觉反应的声音，引起视觉反应的图像等。

参考答案

（12）D

试题（13）

微型计算机系统中，显示器属于（13）。

（13）A. 表现媒体　B. 传输媒体　C. 表示媒体　D. 存储媒体

试题（13）分析

本题考查多媒体的基本知识。

表现媒体是指进行信息输入和输出的媒体，如键盘、鼠标、话筒，以及显示器、打印机、喇叭等；表示媒体指传输感觉媒体的中介媒体，即用于数据交换的编码，如图像编码、文本编码和声音编码等；传输媒体指传输表示媒体的物理介质，如电缆、光缆、电磁波等；存储媒体指用于存储表示媒体的物理介质，如硬盘、光盘等。

参考答案

（13）A

试题（14）

（14）是表示显示器在纵向（列）上具有的像素点数目指标。

（14）A. 显示分辨率　B. 水平分辨率　C. 垂直分辨率　D. 显示深度

试题（14）分析

本题考查多媒体的基本知识。

显示分辨率是指显示器上能够显示出的像素点数目，即显示器在横向和纵向上能够显示出的像素点数目。水平分辨率表明显示器水平方向（横向）上显示出的像素点数目，

垂直分辨率表明显示器垂直方向（纵向）上显示出的像素点数目。例如，显示分辨率为 1024×768 则表明显示器水平方向上显示 1024 个像素点，垂直方向上显示 768 个像素点，整个显示屏就含有 796432 个像素点。屏幕能够显示的像素越多，说明显示设备的分辨率越高，显示的图像质量越高。显示深度是指显示器上显示每个像素点颜色的二进制位数。

参考答案

（14）C

试题（15）

软件工程的基本要素包括方法、工具和__（15）__。

（15）A．软件系统　B．硬件系统　C．过程　D．人员

试题（15）分析

本题考查软件工程的基本概念。

软件工程是一门工程学科，涉及到软件开发的各个方面，从最初的系统描述到交付后的系统维护，都属于其学科范畴。用软件工程方法进行软件开发，涉及到方法、工具和过程等要素。其中，方法是产生某些结果的形式化过程。工具是用更好的方式完成某件事情的设备或自动化系统。过程是把工具和方法结合起来，定义涉及活动、约束和资源使用的一系列步骤，来生产某种想要的输出。

参考答案

（15）C

试题（16）

在__（16）__设计阶段选择适当的解决方案，将系统分解为若干个子系统，建立整个系统的体系结构。

（16）A．概要　B．详细　C．结构化　D．面向对象

试题（16）分析

本题考查软件工程的基本概念。

软件设计的任务是基于需求分析的结果建立各种设计模型，给出问题的解决方案。从工程管理的角度，可以将软件设计分为两个阶段：概要设计阶段和详细设计阶段。结构化设计方法中，概要设计阶段进行软件体系结构的设计、数据设计和接口设计；详细设计阶段进行数据结构和算法的设计。面向对象设计方法中，概要设计阶段进行体系结构设计、初步的类设计/数据设计、结构设计；详细设计阶段进行构件设计。

结构化设计和面向对象设计是两种不同的设计方法，结构化设计根据系统的数据流图进行设计，模块体现为函数、过程及子程序；面向对象设计基于面向对象的基本概念进行，模块体现为类、对象和构件等。

参考答案

（16）A

试题（17）、（18）

某项目包含的活动如下表所示，完成整个项目的最短时间为__（17）__周。不能通过缩短活动__（18）__的工期，来缩短整个项目的完成时间。

活动编号	工期（周）	直接前驱
A	3	-
B	5	A
C	1	B
D	3	A
E	5	D
F	4	C, E
G	3	C, E
H	4	F, G

（17）A．16　　B．17　　C．18　　D．19

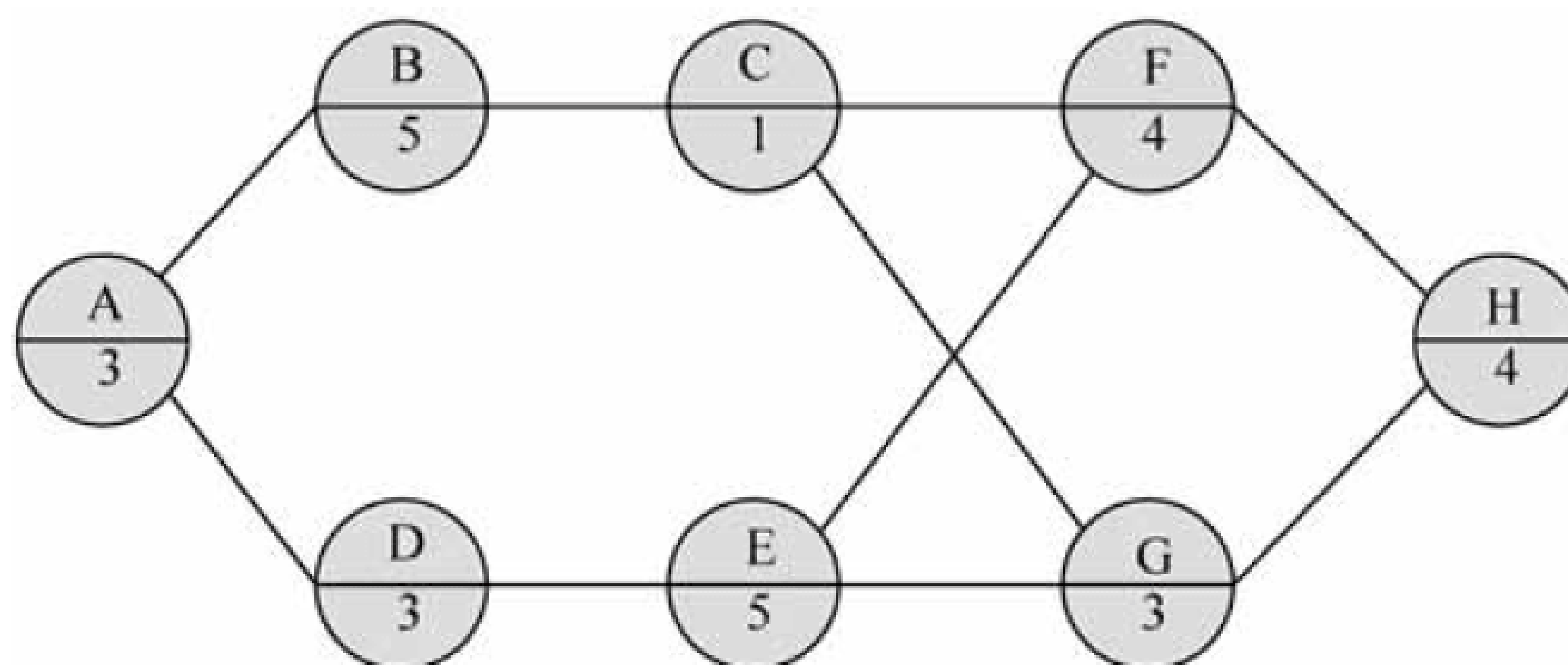
（18）A．A　　B．B　　C．D　　D．F

试题（17）、（18）分析

本题考查软件项目管理的基础知识。

活动图是描述一个项目中各个工作任务相互依赖关系的一种模型，项目的很多重要特性可以通过分析活动图得到，如估算项目完成时间，计算关键路径和关键活动等。

根据上表给出的数据，构建活动图，如下图所示。



从上图很容易看出，关键路径为 A-D-E-F-H，其长度为 19，关键路径上的活动均为关键活动。

参考答案

（17）D　　（18）B

试题（19）

风险的优先级通常是根据__（19）__设定。

（19）A．风险影响（Risk Impact）　　B．风险概率（Risk Probability）

C．风险暴露（Risk Exposure）　　D．风险控制（Risk Control）

试题（19）分析

本题考查软件项目管理的基础知识。

风险是一种具有负面后果的、人们不希望发生的事件。风险管理是软件项目管理的一项重要任务。在进行风险管理时，根据风险的优先级来确定风险控制策略，而优先级是根据风险暴露来确定的。风险暴露是一种量化风险影响的指标，等于风险影响乘以风险概率。风险影响是当风险发生时造成的损失。风险概率是风险发生的可能性。风险控制是风险管理的一个重要活动。

参考答案

（19）C

试题（20）

以下关于程序设计语言的叙述中，错误的是＿（20）＿。

（20）A．程序设计语言的基本成分包括数据、运算、控制和传输等

B．高级程序设计语言不依赖于具体的机器硬件

C．程序中局部变量的值在运行时不能改变

D．程序中常量的值在运行时不能改变

试题（20）分析

本题考查程序语言基础知识。

选项 A 涉及程序语言的一般概念，程序设计语言的基本成分包括数据、运算、控制和传输等。

选项 B 考查高级语言和低级语言的概念。对于程序设计语言，高级语言和低级语言是指其相对于运行程序的机器的抽象程度。低级语言在形式上更接近机器指令，汇编语言就是与机器指令一一对应的。高级语言对底层操作进行了抽象和封装，其一条语句对应多条机器指令，使编写程序的过程更符合人类的思维习惯，并且极大简化了人力劳动。高级语言不依赖于具体的机器硬件。

选项 C 考查局部变量的概念，凡是在函数内部定义的变量都是局部变量（也称作内部变量），包括在函数内部复合语句中定义的变量和函数形参表中说明的形式参数。局部变量只能在函数内部使用，其作用域是从定义位置起至函数体或复合语句体结束为止。局部变量的值通常在其生存期内是变化的。

选项 D 考查常量的概念，程序中常量的值在运行时是不能改变的。

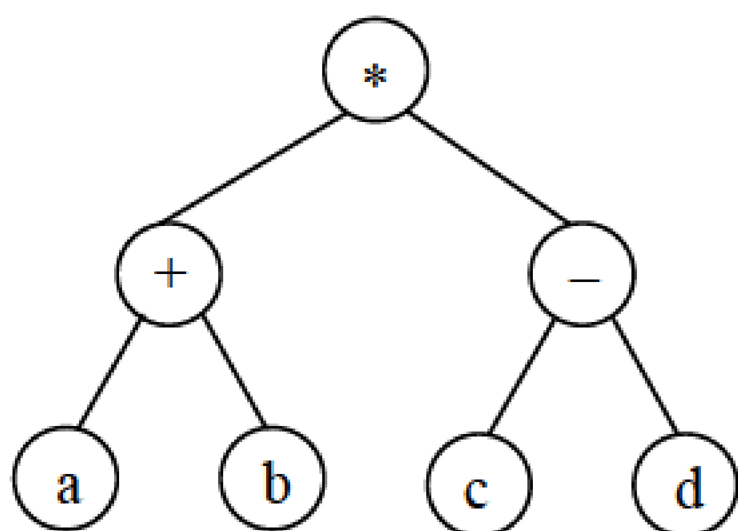
参考答案

（20）C

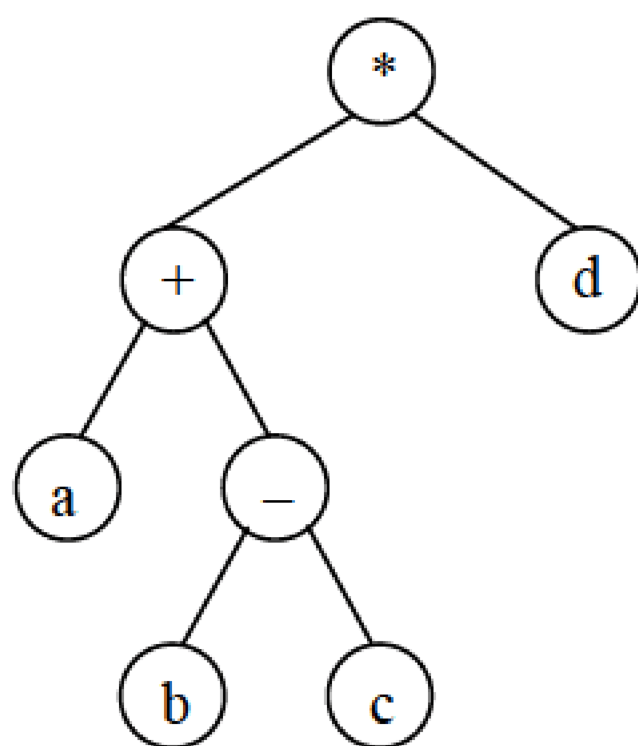
试题（21）

与算术表达式“(a+(b−c))*d”对应的树是＿（21）＿。

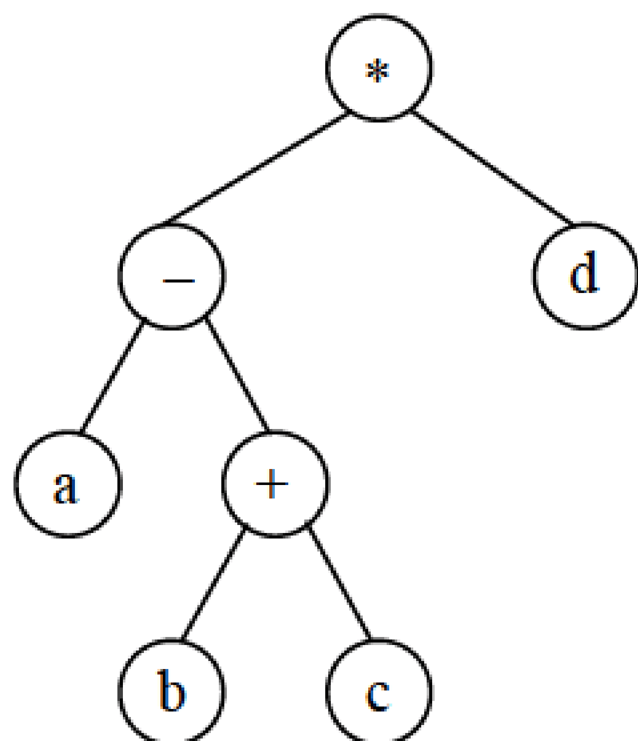
（21）A.



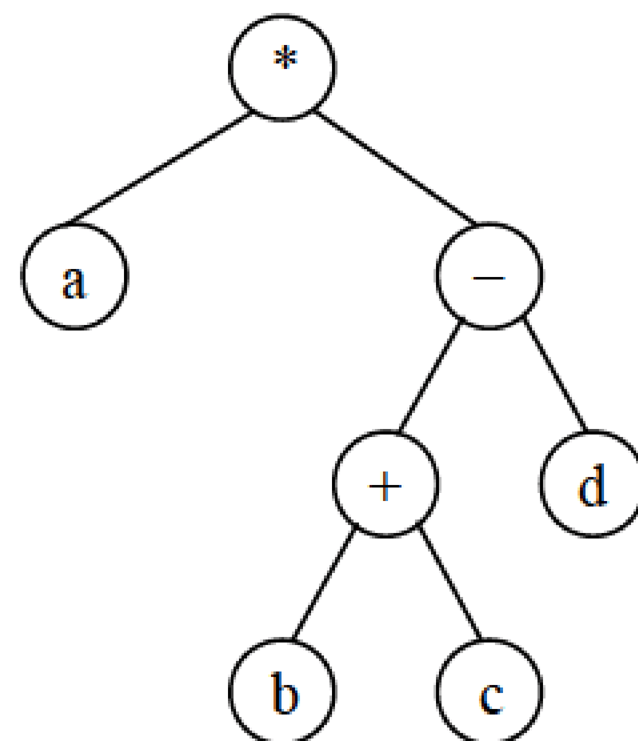
B.



C.



D.



试题（21）分析

本题考查程序语言与数据结构基础知识。

对算术表达式“(a+(b−c))*d”求值的运算处理顺序是：先进行 b−c，然后与 a 相加，最后再与 d 相乘。只有选项 B 所示的二叉树与其相符。

参考答案

（21）B

试题（22）

C 程序中全局变量的存储空间在＿（22）＿分配。

（22）A．代码区　　B．静态数据区　　C．栈区　　D．堆区

试题（22）分析

本题考查程序语言基础知识。

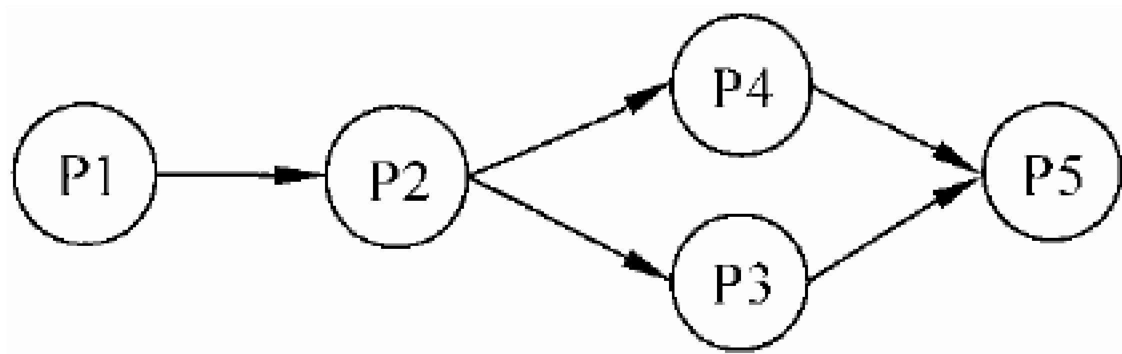
程序运行时的用户内存空间一般划分为代码区、静态数据区、栈区和堆区，其中栈区和堆区也称为动态数据区。全局变量的存储空间在静态数据区。

参考答案

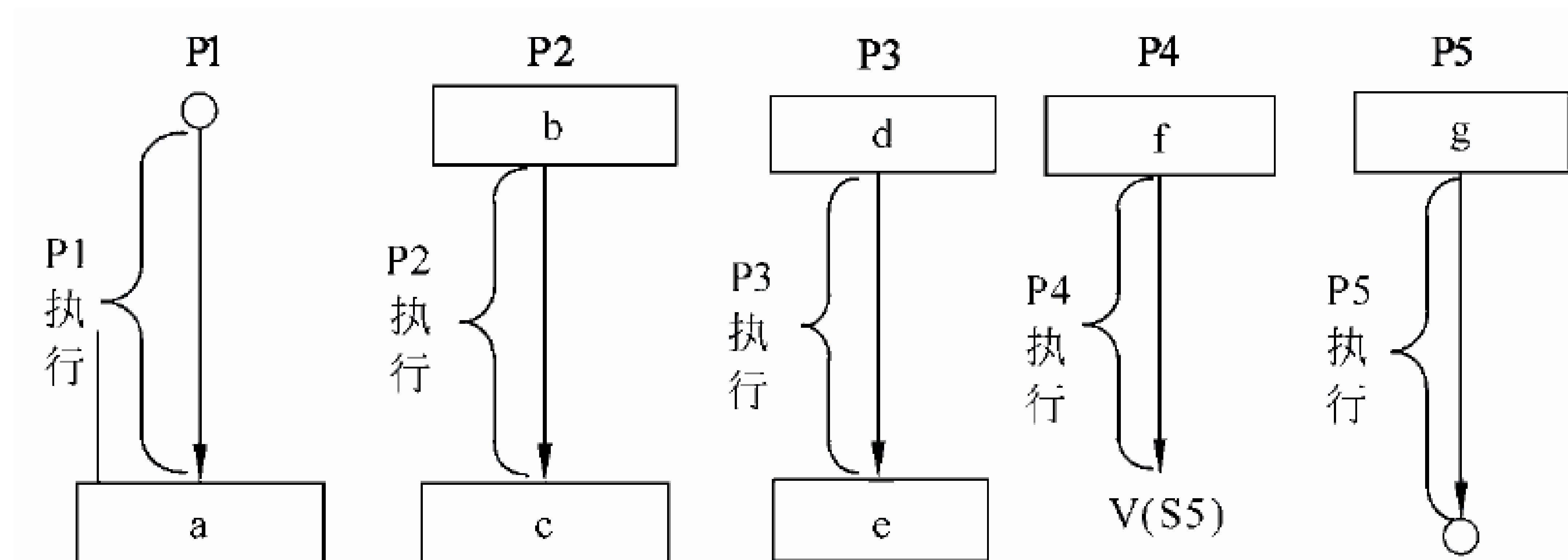
（22）B

试题（23）～（25）

进程 P1、P2、P3、P4 和 P5 的前趋图如下所示：



若用 PV 操作控制进程 P1、P2、P3、P4 和 P5 并发执行的过程，则需要设置 5 个信号量 S1、S2、S3、S4 和 S5，且信号量 S1～S5 的初值都等于零。下图中 a、b 和 c 处应分别填写__（23）__；d 和 e 处应分别填写__（24）__，f 和 g 处应分别填写__（25）__。



（23）A. V（S1）、P（S1）和 V（S2）V（S3）
　　　B. P（S1）、V（S1）和 V（S2）V（S3）
　　　C. V（S1）、V（S2）和 P（S1）V（S3）
　　　D. P（S1）、V（S2）和 V（S1）V（S3）

（24）A. V（S2）和 P（S4）　　B. P（S2）和 V（S4）
　　　C. P（S2）和 P（S4）　　D. V（S2）和 V（S4）

（25）A. P（S3）和 V（S4）V（S5）　　B. V（S3）和 P（S4）P（S5）
　　　C. P（S3）和 P（S4）P（S5）　　D. V（S3）和 V（S4）V（S5）

试题（23）～（25）分析

试题（23）的正确的答案为 A。根据前驱图，P1 进程执行完需要通知 P2 进程，故需要利用 V（S1）操作通知 P2 进程，所以空 a 应填 V（S1）；P2 进程需要等待 P1 进程的结果，故需要利用 P（S1）操作测试 P1 进程是否运行完，所以空 b 应填 P（S1）；又由于 P2 进程运行结束需要利用 V（S2）、V（S3）操作分别通知 P3、P4 进程，所以空 c 应填 V（S2）、V（S3）。

试题（24）的正确的答案为 B。根据前驱图，P3 进程运行前需要等待 P2 进程的结果，故需执行程序前要先利用 1 个 P 操作，根据排除法可选项只有选项 B 和选项 C。又因为 P3 进程运行结束后需要利用 1 个 V 操作通知 P5 进程，根据排除法可选项只有选项 B 满足要求。

试题（25）的正确的答案为 C。根据前驱图，P4 进程执行前需要等待 P2 进程的结果，故空 f 处需要 1 个 P 操作；P5 进程执行前需要等待 P3 和 P4 进程的结果，故空 g 处需要 2 个 P 操作。根据排除法可选项只有选项 C 能满足要求。

参考答案

（23）A　　（24）B　　（25）C

试题（26）

某进程有 4 个页面，页号为 0～3，页面变换表及状态位、访问位和修改位的含义如下图所示。若系统给该进程分配了 3 个存储块，当访问的页面 1 不在内存时，淘汰表中页号为__（26）__的页面代价最小。

页号	页帧号	状态位	访问位	修改位
0	6	1	1	1
1	—	0	0	0
2	3	1	1	1
3	2	1	1	0

状态位含义 {=0 不在内存；=1 在内存}
访问位含义 {=0 未访问过；=1 访问过}
修改位含义 {=0 未修改过；=1 修改过}

（26）A．0　　B．1　　C．2　　D．3

试题（26）分析

试题（26）的正确选项为 D。根据题意，页面变换表中状态位等于 0 和 1 分别表示页面不在内存或在内存，所以 0、2 和 3 号页面在内存。当访问的页面 1 不在内存时，系统应该首先淘汰未被访问的页面，因为根据程序的局部性原理最近未被访问的页面下次被访问的概率更小；如果页面最近都被访问过，应该先淘汰未修改过的页面。因为未修改过的页面内存与辅存一致，故淘汰时无须写回辅存，使系统页面置换代价小。经上述分析，0、2 和 3 号页面都是最近被访问过的，但 0 和 2 号页面都被修改过而 3 号页面未修改过，故应该淘汰 3 号页面。

参考答案

（26）D

试题（27）、（28）

嵌入式系统初始化过程主要有 3 个环节，按照自底向上、从硬件到软件的次序依次为__（27）__。系统级初始化主要任务是__（28）__。

（27）A．片级初始化→系统级初始化→板级初始化
B．片级初始化→板级初始化→系统级初始化
C．系统级初始化→板级初始化→片级初始化
D．系统级初始化→片级初始化→板级初始化

（28）A．完成嵌入式微处理器的初始化
B．完成嵌入式微处理器以外的其他硬件设备的初始化
C．以软件初始化为主，主要进行操作系统的初始化
D．设置嵌入式微处理器的核心寄存器和控制寄存器工作状态

试题（27）、（28）分析

本题考查应试者对嵌入式系统方面的基础知识。

试题（27）的正确选项为 B，试题（28）的正确选项为 C。嵌入式系统初始化过程可以分为 3 个主要环节，按照自底向上、从硬件到软件的次序依次为：片级初始化、板级初始化和系统级初始化。

片级初始化完成嵌入式微处理器的初始化，包括设置嵌入式微处理器的核心寄存器和控制寄存器、嵌入式微处理器核心工作模式和嵌入式微处理器的局部总线模式等。片级初始化把嵌入式微处理器从上电时的默认状态逐步设置成系统所要求的工作状态。这是一个纯硬件的初始化过程。

板级初始化完成嵌入式微处理器以外的其他硬件设备的初始化。另外，还需设置某些软件的数据结构和参数，为随后的系统级初始化和应用程序的运行建立硬件和软件环境。这是一个同时包含软硬件两部分在内的初始化过程。

系统初始化过程以软件初始化为主，主要进行操作系统的初始化。BSP 将对嵌入式微处理器的控制权转交给嵌入式操作系统，由操作系统完成余下的初始化操作，包含加载和初始化与硬件无关的设备驱动程序，建立系统内存区，加载并初始化其他系统软件模块，如网络系统、文件系统等。最后，操作系统创建应用程序环境，并将控制权交给应用程序的入口。

参考答案

（27）B　　　（28）C

试题（29）

某公司计划开发一个产品，技术含量很高，与客户相关的风险也很多，则最适于采用<u>（29）</u>开发过程模型。

（29）A．瀑布　　　B．原型　　　C．增量　　　D．螺旋

试题（29）分析

本题考查软件过程模型的基础知识。

瀑布模型将软件生存周期各个活动规定为线性顺序连接的若干阶段的模型，规定了由前至后，相互衔接的固定次序，如同瀑布流水，逐级下落。这种方法是一种理想的开发模式，缺乏灵活性，特别是无法解决软件需求不明确或不准确的问题。

原型模型从初始的原型逐步演化成最终软件产品，特别适用于对软件需求缺乏准确认识的情况。

增量开发是把软件产品作为一系列的增量构件来设计、编码、集成和测试，可以在增量开发过程中逐步理解需求。

螺旋将瀑布模型与快速原型模型结合起来，并且加入两种模型均忽略了的风险分析，适用于复杂的大型软件。

参考答案

（29）D

试题（30）

在敏捷过程的方法中，__（30）__认为每一个不同的项目都需要一套不同的策略、约定和方法论。

（30）A．极限编程（XP） B．水晶法（Crystal）
C．并列争球法（Scrum） D．自适应软件开发（ASD）

试题（30）分析

本题考查敏捷方法的基础知识。

在 20 世纪 90 年代后期，一些开发人员抵制严格化软件开发过程，试图强调灵活性在快速有效的软件生产中的作用，提出了敏捷宣言，即个人和交互胜过过程和工具；可以运行的软件胜过面面俱到的文档；与客户合作胜过合同谈判；对变化的反应胜过遵循计划。基于这些基本思想，有很多敏捷过程的典型方法，其中：

极限编程 XP 是激发开发人员创造性、使得管理负担最小的一组技术。

水晶法 Crystal 认为每一个不同的项目都需要一套不同的策略、约定和方法论。

并列争球法（Scrum）使用迭代的方法，其中把每 30 天一次的迭代称为一个冲刺，并按需求的优先级来实现产品。多个自组织和自治小组并行地递增实现产品，协调是通过简短的日常情况会议进行。

自适应软件开发（ASD）有六个基本的原则：

① 在自适应软件开发中，有一个使命作为指导，它设立了项目的目标，但并不描述如何达到这个目标；

② 特征被视为客户键值的关键，因此，项目是围绕着构造的构件来组织并实现特征；

③ 过程中的迭代是很重要的，因此重做与做同样重要，变化也包含其中；

④ 变化不视为是一种更正，而是对软件开发实际情况的调整；

⑤ 确定的交付时间迫使开发人员认真考虑每一个生产版本的关键需求；

⑥ 风险也包含其中，它使开发人员首先跟踪最艰难的问题。

参考答案

（30）B

试题（31）

软件配置管理的内容不包括__（31）__。

（31）A．版本控制 B．变更控制 C．过程支持 D．质量控制

试题（31）分析

本题考查软件配置管理的基础知识。

软件配置管理 SCM 用于整个软件工程过程，其主要目标是标识变更、控制变更、确保变更正确的实现，报告变更。其主要内容包括版本管理、配置支持、变更支持、过程支持、团队支持、变化报告和审计支持等。

参考答案

（31）D

试题（32）

某模块实现两个功能：向某个数据结构区域写数据和从该区域读数据。该模块的内聚类型为__（32）__内聚。

（32）A．过程　　B．时间　　C．逻辑　　D．通信

试题（32）分析

本题考查软件设计的基础知识。

模块独立性是创建良好设计的一个重要原则，一般采用模块间的耦合和模块的内聚两个准则来进行度量。耦合程度越低，内聚程度越高，则模块的独立性越好。

存在多种模块内聚类型，从高到低依次为功能、通信、顺序、过程、时间、逻辑和偶然内聚。

一个模块内的几个操作是相关的，而且必须以特定的次序执行，则该模块的内聚类型为过程内聚。

模块的各个功能的执行与时间有关，通常要求所有功能必须在同一时间段内执行，则该模块的内聚类型为时间内聚。

若一个模块的各个部分只是通过代码的逻辑结构相关联，则该模块的内聚类型为逻辑内聚。

把访问或操作在同一数据结构的操作放在一个模块中，则该模块的内聚类型为通信内聚。

参考答案

（32）D

试题（33）

正式技术评审的目标是__（33）__。

（33）A．允许高级技术人员修改错误
B．评价程序员的工作效率
C．发现软件中的错误
D．记录程序员的错误情况并与绩效挂钩

试题（33）分析

本题考查软件评审的基础知识。

软件评审的内容包括设计质量评审、程序质量评审和与运行环境接口的评审。评审的主要目标是为了发现软件中的错误。

参考答案

（33）C

试题（34）

自底向上的集成测试策略的优点包括__（34）__。

（34）A．主要的设计问题可以在测试早期处理

B．不需要写驱动程序

C．不需要写桩程序

D．不需要进行回归测试

试题（34）分析

本题考查软件测试的基础知识。

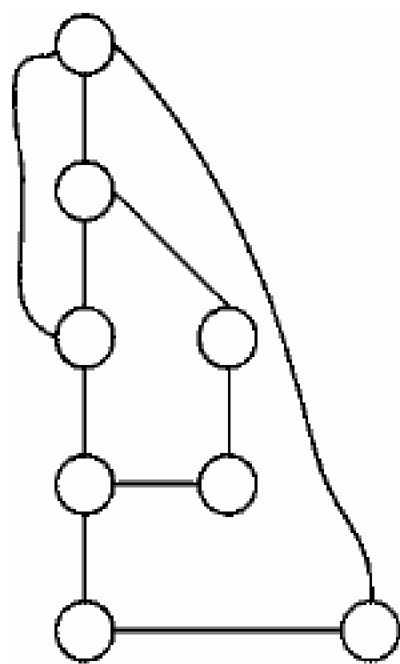
软件测试分为单元测试、集成测试和系统测试。集成测试把模块按系统设计说明书的要求组合起来进行测试。存在多种组合的集成测试策略：自底向上、自顶向下、一次性集成、三明治集成等。其中，自底向上策略通过不断合并底层模块来测试更高层模块的方式进行。在该方法中，需要写驱动程序来调用待测试的底层模块，主要的设计问题需要到测试后期才能发现。由于从最底层的模块开始测试，因此过程中不需要写桩模块。当测试过程中发现错误时，需要进行回归测试。

参考答案

（34）C

试题（35）

采用 McCabe 度量法计算下列程序图的环路复杂性为__（35）__。



（35）A．2　　B．3　　C．4　　D．5

试题（35）分析

本题考查软件度量的基础知识。

软件复杂性主要表现在程序的复杂性，而程序的复杂性主要指模块内程序的复杂性。McCabe 度量法是一种度量程序复杂性的方法，该方法用程序图来表示实际的程序，如上图所示，用环路数来表示程序复杂性，即 $V(G) = m - n + 2p$，其中 m 和 n 分别表示图中的边数和顶点数，p 为 1。本题中，m=10，n=8，因此 $V(G) = 10-8+2=4$。

参考答案

（35）C

试题（36）

以下关于软件可维护性的叙述中，不正确的是“可维护性__（36）__”。

（36）A. 是衡量软件质量的一个重要特性
B. 不受软件开发文档的影响
C. 是软件开发阶段各个时期的关键目标
D. 可以从可理解性、可靠性、可测试性、可行性、可移植性等方面进行度量

试题（36）分析

本题考查维护方面的基础知识。

软件交付给用户使用后到软件报废之前都属于软件维护阶段。软件系统的可维护性可以定义为：维护人员理解、改正、改动和改进该软件的难易程度。提供软件可维护性是开发软件系统所有步骤的关键目的，是衡量软件质量的一种重要特性，可以从可理解性、可靠性、可测试性、可行性、可移植性等方面进行度量。良好的软件开发文档可以有效地提高软件的可维护性。

参考答案

（36）B

试题（37）

对象、类、继承和消息传递是面向对象的 4 个核心概念。其中对象是封装__（37）__的整体。

（37）A. 命名空间　B. 要完成任务　C. 一组数据　D. 数据和行为

试题（37）分析

本题考查面向对象的基本知识。

面向对象的 4 个核心概念是对象、类、继承和消息传递。其中，对象是基本的运行时的实体，它既包括数据（属性），也包括作用于数据的操作（行为）。所以，一个对象把属性和行为封装为一个整体。类定义了一组大体上相似的对象。一个类所包含的方法和数据描述一组对象的共同行为和属性。在进行类设计时，有些类之间存在一般和特殊关系，即一些类是某个类的特殊情况，某个类是一些类的一般情况，这就是继承关系。消息是对象之间进行通信的一种构造，包含要求接收对象去执行某些活动的信息。

参考答案

（37）D

试题（38）

面向对象__（38）__选择合适的面向对象程序设计语言，将程序组织为相互协作的对象集合，每个对象表示某个类的实例，类通过继承等关系进行组织。

（38）A. 分析　B. 设计　C. 程序设计　D. 测试

试题（38）分析

本题考查面向对象的基本知识。

在采用面向对象技术开发系统时，主要步骤有面向对象分析、面向对象设计、面向对象程序设计和面向对象测试。面向对象分析主要包括：认定对象、组织对象、描述对象间的相互作用、定义对象的操作、定义对象的内部信息。面向对象设计是设计分析模型和实现相应源代码。面向对象程序设计选择合适的面向对象程序设计语言，将程序组织为相互协作的对象集合，每个对象表示某个类的实例，类通过继承等关系进行组织。面向对象测试是尽可能早的开始进行系统测试，以发现系统中可能存在的错误并进行修复，进而保证系统质量。

参考答案

（38）C

试题（39）

一个类可以具有多个同名而参数类型列表不同的方法，被称为方法__（39）__。

（39）A. 重载　　B. 调用　　C. 重置　　D. 标记

试题（39）分析

本题考查面向对象的基本知识。

在面向对象技术中，一个类可以具有多个同名而参数类型列表不同的方法，每个方法实现自己的特定行为，这种机制称为方法重载（Overloading）。对象协作过程中，对象给对象进行消息传递，通常通过方法调用进行。在继承关系中，子类重新定义父类中已经定义的方法，称为方法重置或覆盖（Overriding）。

参考答案

（39）A

试题（40）、（41）

UML 中有 4 种关系：依赖、关联、泛化和实现。__（40）__是一种结构关系，描述了一组链，链是对象之间的连接；__（41）__是一种特殊/一般关系，使子元素共享其父元素的结构和行为。

（40）A. 依赖　　B. 关联　　C. 泛化　　D. 实现

（41）A. 依赖　　B. 关联　　C. 泛化　　D. 实现

试题（40）、（41）分析

本题考查统一建模语言（UML）的基本知识。

UML 是面向对象软件的标准化建模语言，由三个要素构成：UML 的基本构造块、支配这些构造块如何放置在一起的规则和运用与整个语言的一些公共机制。UML 的词汇表包含三种构造块：事物、关系和图。事物是对模型中最具有代表性的成分的抽象；关系把事物结合在一起；图聚集了相关的事物。其中关系包括 4 种：依赖、关联、泛化和实现。

依赖（Dependency）是两个事物间的语义关系，其中一个事物（独立事物）发生变化会影响另一个事物（依赖事物）的语义。关联（Association）是一种结构关系，它描述了一组链，链是对象之间的连接。泛化（Generalization）是一种特殊/一般关系，特殊元素（子元素）的对象可替代一般元素（父元素）的对象，用这种方法，子元素共享了父元素的结构和行为。实现（Realization）是类元之间的语义关系，其中一个类元指定了由另一个类元保证执行的契约。

参考答案

（40）B　　（41）C

试题（42）、（43）

UML 图中，对新开发系统的需求进行建模，规划开发什么功能或测试用例，采用 （42） 最适合。而展示交付系统的软件组件和硬件之间的关系的图是 （43） 。

（42）A. 类图　　B. 对象图　　C. 用例图　　D. 交互图

（43）A. 类图　　B. 部署图　　C. 组件图　　D. 网络图

试题（42）、（43）分析

本题考查统一建模语言（UML）的基本知识。

UML 中提供了多种建模系统需求的图，体现系统的静态方面和动态方面。

类图（Class Diagram）展现了一组对象、接口、协作和它们之间的关系。在面向对象系统的建模中，最常见的就是类图，它给出系统的静态设计视图。对象图（Object Diagram）展现了某一时刻一组对象以及它们之间的关系。对象图描述了在类图中所建立的事物的实例的静态快照，给出系统的静态设计视图或静态进程视图。用例图（Use Case Diagram）展现了一组用例、参与者（Actor）以及它们之间的关系。这个视图主要支持系统的行为，即该系统在它的周边环境的语境中所提供的外部可见服务。用例图用于对一个系统的需求进行建模，包括说明这个系统应该做什么（从系统外部的一个视点出发），而不考虑系统应该怎样做。交互图用于对系统的动态方面进行建模。一张交互图表现的是一个交互，由一组对象和它们之间的关系组成，包含它们之间可能传递的消息。交互图表现为序列图、通信图、交互概览图和时序图，每种针对不同的目的，能适用于不同的情况。序列图是强调消息时间顺序的交互图；通信图是强调接收和发送消息的对象的结构组织的交互图；交互概览图强调控制流的交互图。时序图（Timing Diagram）关注沿着线性时间轴、生命线内部和生命线之间的条件改变。

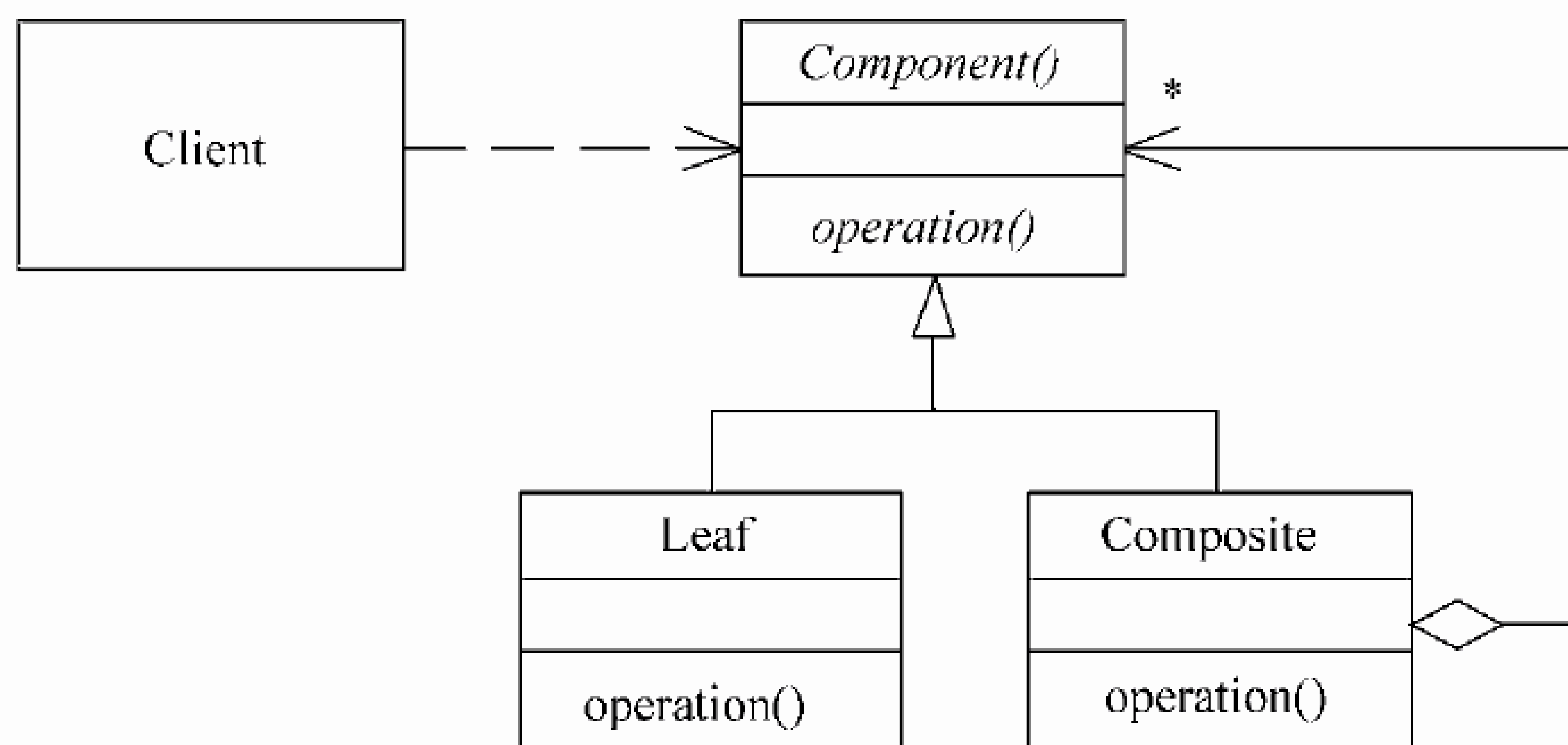
部署图（Deployment Diagram）是用来对面向对象系统的物理方面建模的方法，展现了运行时处理结点以及其中构件（制品）的配置。组件图（Component Diagram）展现了一组组件之间的组织和依赖。

参考答案

（42）C （43）B

试题（44）～（46）

下图所示为__（44）__设计模式，属于__（45）__设计模式，适用于__（46）__。



（44）A．代理（Proxy） B．生成器（Builder）

C．组合（Composite） D．观察者（Observer）

（45）A．创建型 B．结构型 C．行为 D．结构型和行为

（46）A．表示对象的部分-整体层次结构时

B．当一个对象必须通知其他对象，而它又不能假定其他对象是谁时

C．当创建复杂对象的算法应该独立于该对象的组成部分及其装配方式时

D．在需要比较通用和复杂的对象指针代替简单的指针时

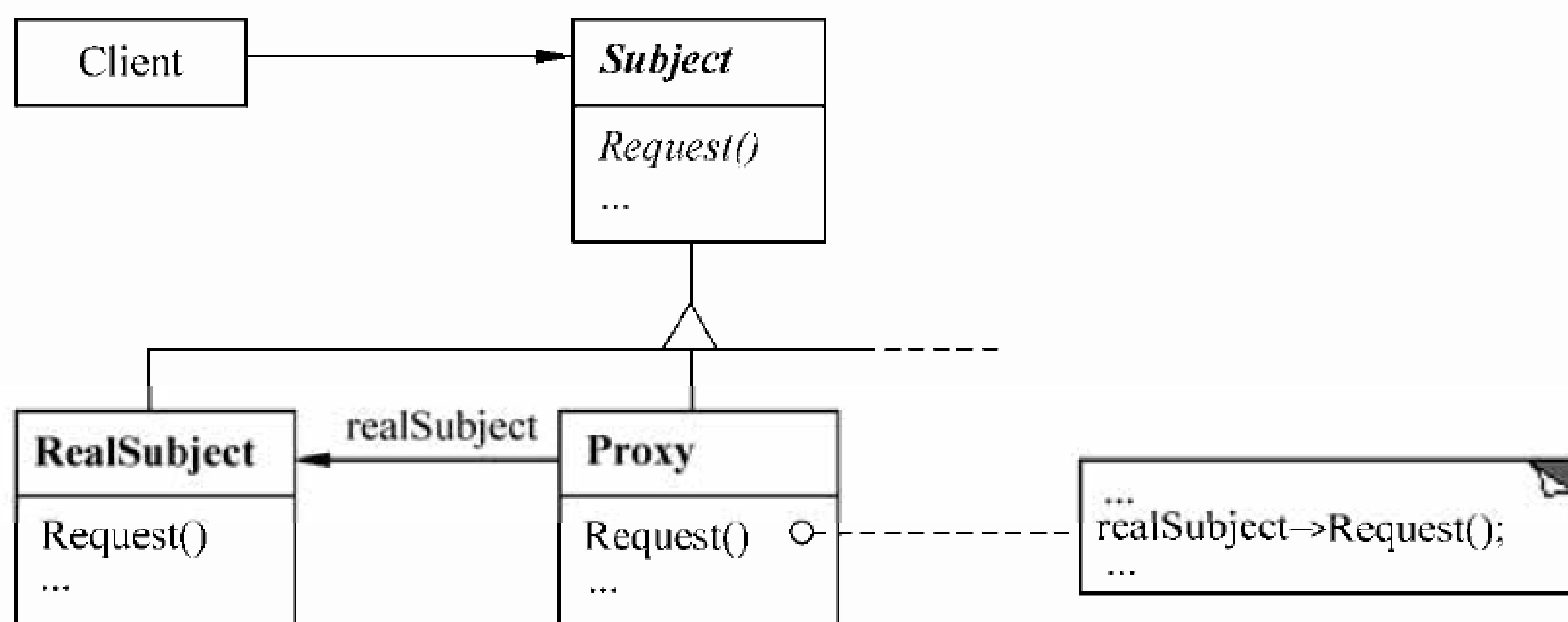
试题（44）～（46）分析

本题考查设计模式的基本概念。

每一个设计模式都集中于一个特定的面向对象设计问题或设计要点，描述了什么时候使用它，在另一些设计约束条件下是否还能使用，以及使用的效果和如何取舍。按照设计模式的目的可以分为创建型、结构型和行为型三大类。创建型模式与对象的创建有关；结构型模式处理类或对象的组合；行为型模式对类或对象怎样交互和怎样分配职责进行描述。每种设计模式都有其适应性，描述适用于解决的问题场合。

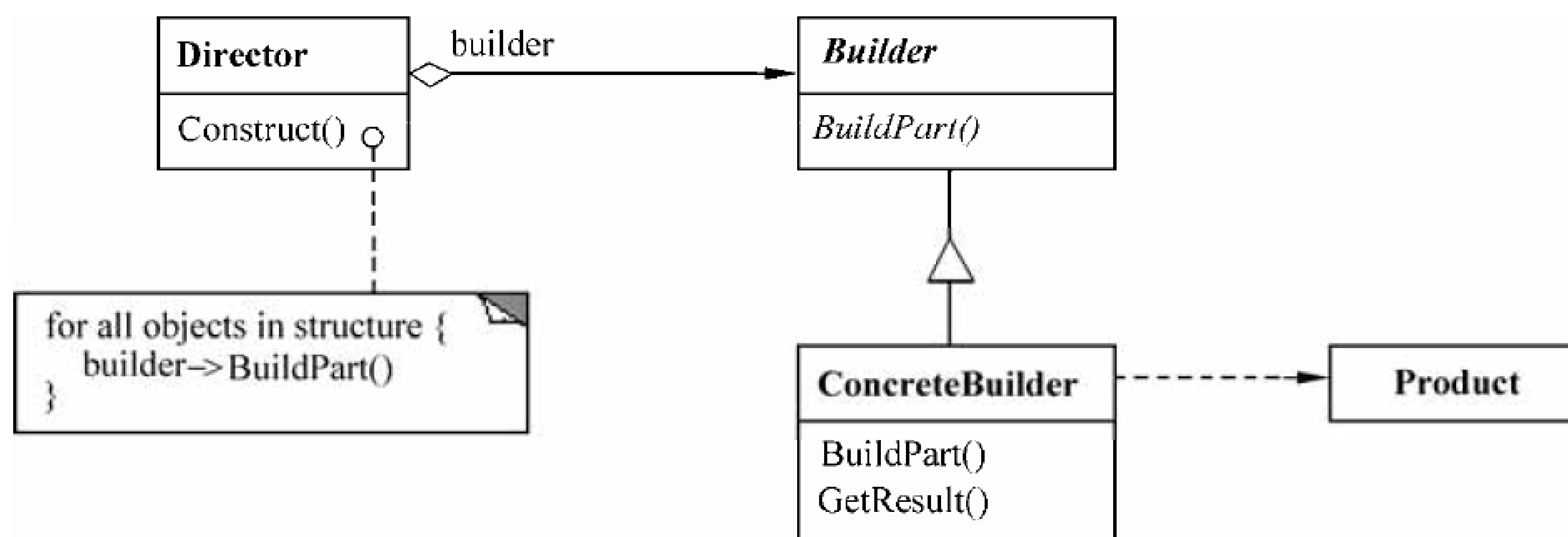
创建型模式包括 Factory Method、Abstract Factory、Builder、Prototype 和 Singleton。结构型模式包括 Adapter（类）、Adapter（对象）、Bridge、Composite、Decorator、Façade、Flyweight 和 Proxy。行为型模式包括 Interpreter、Template Method、Chain of Responsibility、Command、Iterator、Mediator、Memento Observer State Strategy 和 Visitor。

Proxy 模式的结构图如下所示：



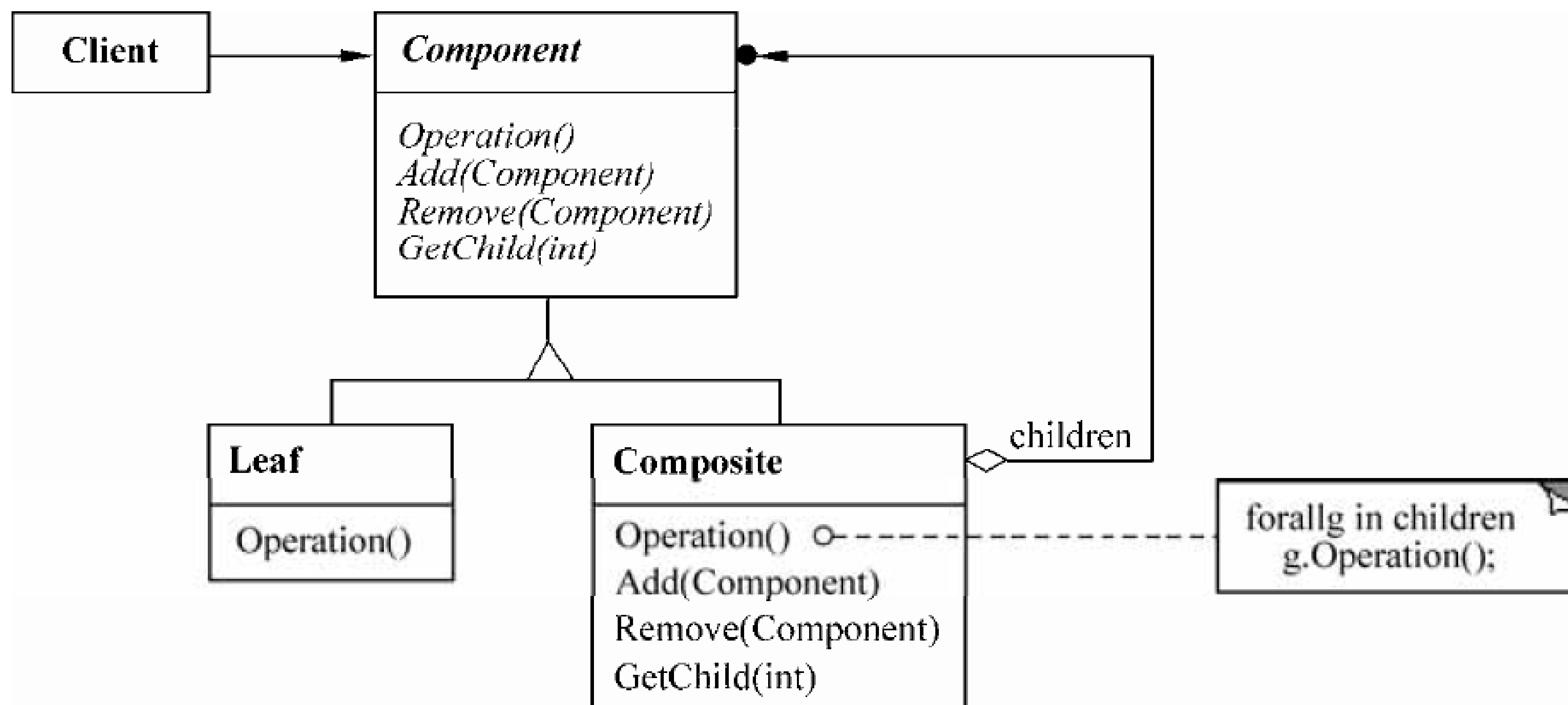
Proxy 模式适用于在需要比较通用和复杂的对象指针代替简单的指针的时候，常见情况有：远程代理（Remote Proxy）为一个对象在不同地址空间提供据不代表；虚代理（Virtual Proxy）根据需要创建开销很大的对象；保护代理（Protection Proxy）控制对原始对象的访问，用于对象应该有不同的访问权限的时候；智能指引（Smart Reference）取代了简单的指针，它在访问对象时执行一些附加操作。

Builder 模式的结构图如下所示：



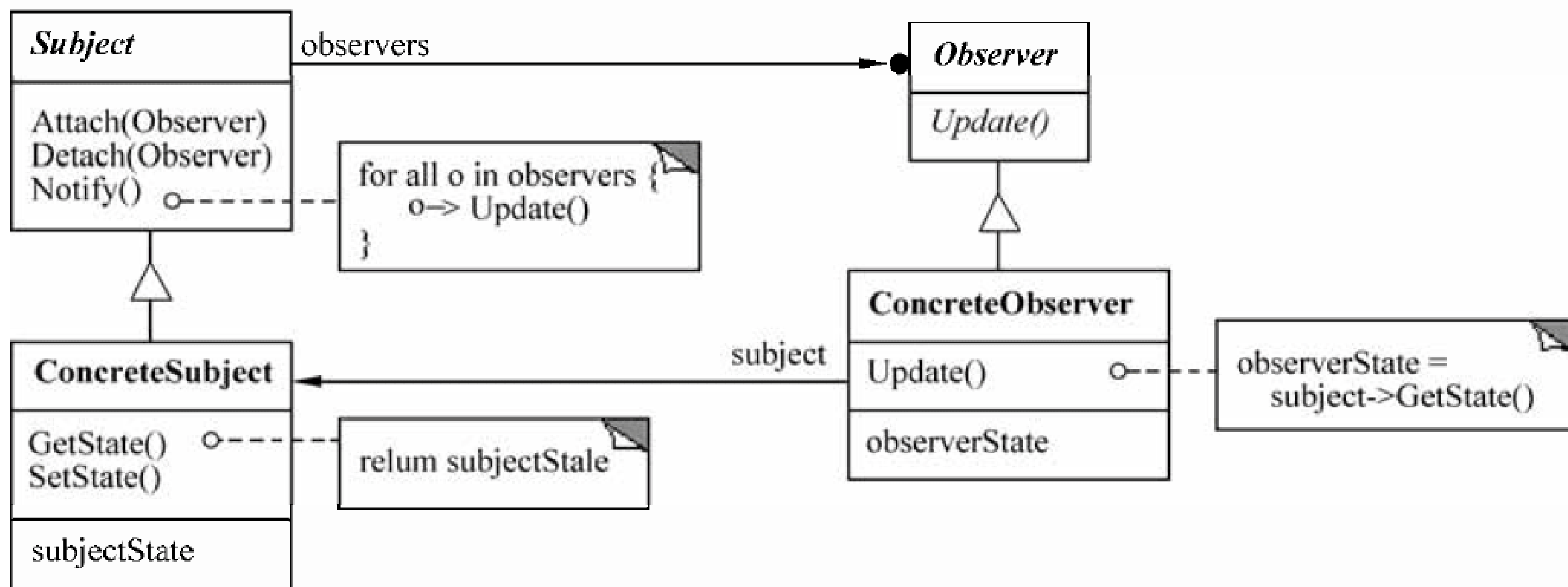
Builder 模式适用于当创建复杂对象的算法应该独立于该对象的组成部分以及它们的装配方式时；当构造过程必须允许被构造的对象有不同的表示时。

Composite 模式的结构图如下所示：



Composite 模式适用于：想表示对象的部分-整体层次结构；希望用户忽略组合对象与单个对象的不同，用户将统一地使用组合结构中的所有对象。

Observer 模式的结构图如下所示：



Observer 模式适用于：当一个抽象模型有两个方面，其中一个方面依赖于另一个方面。将这两者封装在独立地对象中以使它们可以各自独立地改变和复用；当对一个对象的改变需要同时改变其他对象，而不知道具体有多少对象有待改变时；当一个对象必须通知其他对象，而它又不能假定其他对象是谁，即不希望这些对象是紧耦合的。

参考答案

（44）C　　（45）B　　（46）A

试题（47）

某些设计模式会引入总是被用作参数的对象。例如__（47）__对象是一个多态的 accept 方法的参数。

（47）A．Visitor　　B．Command　　C．Memento　　D．Observer

试题（47）分析

本题考查设计模式的概念，对行为模式进行比较。

很多行为模式注重封装变化。当一个程序的某个方面的特征经常发生改变时，这些模式就定义一个封装这个方面的对象。这样，当该程序的其他部分依赖于这个方面时，它们都可以与此对象协作。一些模式引入总是被用作参数的对象。有些模式定义一些可作为令牌进行传递的对象，这些对象将在稍后被调用。

在 Visitor 模式中，一个 Visitor 对象是一个多态的 accept 操作的参数，这个操作作用于该 Visitor 对象访问的对象。在 Command 模式中，令牌代表一个请求；在 Memento 模式中，它代表在一个对象在某个特定时刻的内部状态。在这两种情况下，令牌都可以有一个复杂的内部表示，但客户并不会意识到这一点。在 Observer 模式中，通过引入 Observer 和 Subject 对象来分布通信。

参考答案

（47）A

试题（48）

对高级语言源程序进行编译或解释的过程可以分为多个阶段，解释方式不包含__（48）__阶段。

（48）A．词法分析　　B．语法分析　　C．语义分析　　D．目标代码生成

试题（48）分析

本题考查程序语言基础知识。

用某种高级语言或汇编语言编写的程序称为源程序，源程序不能直接在计算机上执行。汇编语言源程序需要用一个汇编程序将其翻译成目标程序后才能执行。高级语言源程序则需要对应的解释程序或编译程序对其进行翻译，然后在机器上运行。

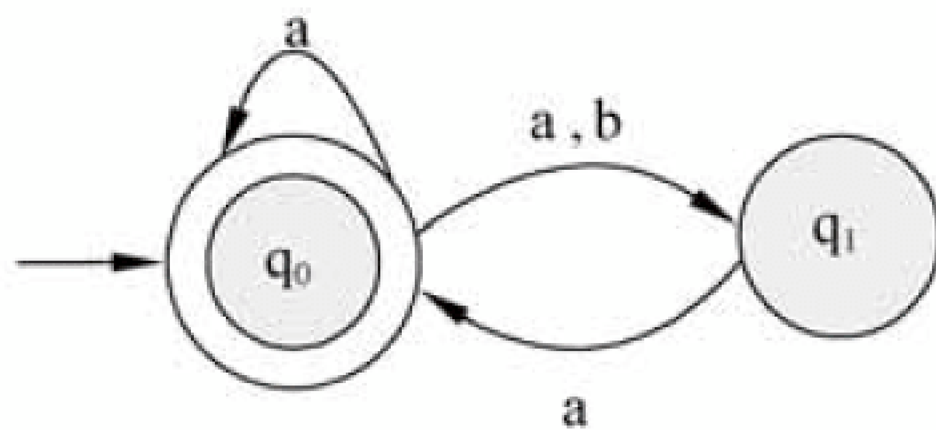
解释程序也称为解释器，它或者直接解释执行源程序，或者将源程序翻译成某种中间代码后再加以执行；而编译程序（编译器）则是将源程序翻译成目标语言程序，然后在计算机上运行目标程序。这两种语言处理程序的根本区别是：在编译方式下，机器上运行的是与源程序等价的目标程序，源程序和编译程序都不再参与目标程序的执行过程；而在解释方式下，解释程序和源程序（或其某种等价表示）要参与到程序的运行过程中，运行程序的控制权在解释程序。简单来说，在解释方式下，翻译源程序时不生成独立的目标程序，而编译器则将源程序翻译成独立保存的目标程序。

参考答案

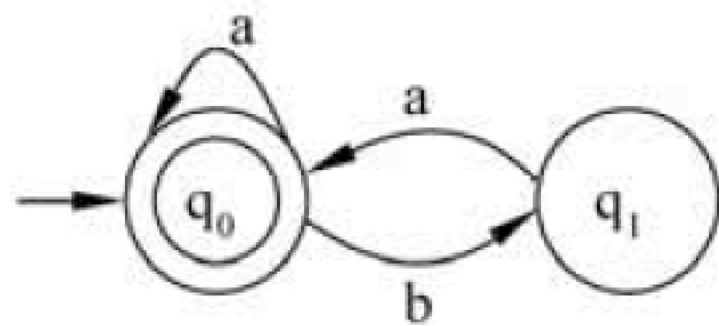
（48）D

试题（49）

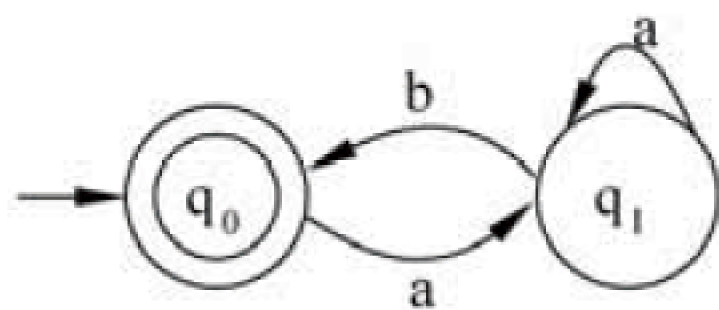
某非确定的有限自动机（NFA）的状态转换图如下图所示（q_0 既是初态也是终态），与该 NFA 等价的确定的有限自动机（DFA）是__（49）__。



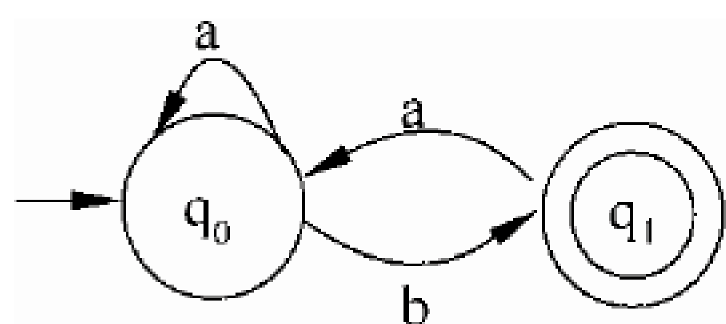
（49）A.



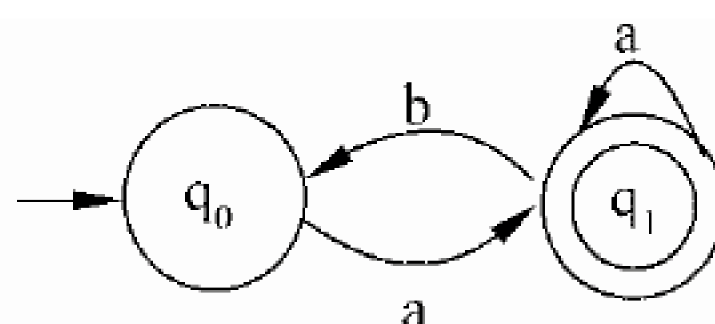
B.



C.



D.



试题（49）分析

本题考查程序语言基础知识。

对高级语言源程序 进行编译（或解释）方式的翻译过程中，词法分析采用有限自动机作为计算模型。有限自动机分为确定的有限自动机（DFA）和不确定的有限自动机（NFA）。可将一个从 NFA 转换为等价的最小化 DFA。

题中的 NFA 的功能是识别空串以及 b 不能连续出现（即每个 b 后至少含有 1 个 a）的 a、b 字符串，若是非空串，则以 a 结尾。

选项 A 识别的是空串以及每个 b 后至少含有 1 个 a 的 a、b 字符串，若是非空串，则以 a 结尾。

选项 B 识别空串以及 b 不能连续出现且以 b 结尾的 a、b 字符串。

选项 C 识别 b 不能连续出现且以 b 结尾的 a、b 字符串，不能识别空串。

选项 D 识别 b 不能连续出现且以 a 结尾的 a、b 字符串，不能识别空串。

参考答案

（49）A

试题（50）

递归下降分析方法是一种＿（50）＿方法。

（50）A. 自底向上的语法分析　　B. 自上而下的语法分析
　　　C. 自底向上的词法分析　　D. 自上而下的词法分析

试题（50）分析

本题考查程序语言基础知识。

对高级语言源程序进行编译（或解释）方式的翻译过程中，语法分析的任务是根据语言的语法规则，分析单词串是否构成短语和句子，即表达式、语句和程序等基本语言结构，同时检查和处理程序中的语法错误。程序设计语言的绝大多数语法规则可以采用上下文无关文法进行描述。语法分析方法有多种，根据产生语法树的方向，可分为自底向上和自顶向下两类。递归下降分析法和预测分析法是常用的自顶向下分析法。算符优先分析法和 LR 分析法属于自底向上的语法分析方法。

参考答案

（50）B

试题（51）

若关系 R（H，L，M，P）的主键为全码（All-key），则关系 R 的主键应＿（51）＿。

（51）A. 为 HLMP
B. 在集合{H，L，M，P}中任选一个
C. 在集合{H L，HM，HP，LM，LP，MP}中任选一个
D. 在集合{H LM，HLP，HMP，LMP }中任选一个

试题（51）分析

本题考查关系数据库系统中关系的键的基本概念。

在关系数据库系统中，全码（All-key）是指关系模型的所有属性组是这个关系模式的候选键，本题所有属性组为 HLMP，故本题的正确选项为 A。

参考答案

（51）A

试题（52）

给定关系模式 $R(A_1,A_2,A_3,A_4)$ 上的函数依赖集 $F=\{A_1A_3 \to A_2, A_2 \to A_3\}$。若将 R 分解为 $\rho=\{(A_1,A_2),(A_1,A_3)\}$，则该分解是__（52）__的。

（52）A. 无损联接且不保持函数依赖　　B. 无损联接且保持函数依赖
C. 有损联接且保持函数依赖　　D. 有损联接且不保持函数依赖

试题（52）分析

本题考查关系数据库规范化理论方面的基础知识。

试题（52）正确答案为 D。设 $U1=\{A_1,A_2\}$，$U2=\{A_1,A_3\}$，那么可得出：$(U1 \cap U2) \to (U1-U2)=A_1 \to A_2$，$(U1 \cap U2) \to (U2-U1)=A_1 \to A_3$，

而 $A_1 \to A_2, A_1 \to A_3 \notin F^+$，所以分解 ρ 是有损连接的。

又因为 $F1=F2=\phi$，$F^+ \neq (F1 \cup F2)^+$，所以分解不保持函数依赖。

参考答案

（52）D

试题（53）

__（53）__算法采用模拟生物进化的三个基本过程“繁殖（选择）→交叉（重组）→变异（突变）”。

（53）A. 粒子群　　B. 人工神经网络　　C. 遗传　　D. 蚁群

试题（53）分析

本题考查数据挖掘基本概念方面的基础知识。

数据挖掘就是应用一系列技术从大型数据库或数据仓库中提取人们感兴趣的信息和知识，这些知识或信息是隐含的，事先未知而潜在有用的，提取的知识表示为概念、规则、规律、模式等形式。也可以说，数据挖掘是一类深层次的数据分析。常见和应用最广泛的数据挖掘方法如下。

① 决策树：决策树方法是利用信息论中的互信息（信息增益）寻找数据库中具有最大信息量的属性字段，建立决策树的一个结点，再根据该属性字段的不同取值建设树

的分支；在每个分支子集中重复建立树的下层结点和分支的过程。国际上最早的、也是最有影响的决策树方法是 Quiulan 研究的 ID3 方法。

② 神经网络：神经网络方法是模拟人脑神经元结构，完成类似统计学中的判别、回归、聚类等功能，是一种非线性的模型，主要有三种神经网络模型：前馈式网络、反馈式网络和自组织网络。人工神经网络最大的长处是可以自动地从数据中学习，形成知识，这些知识有些是我们过去未曾发现的，因此它具有较强的创新性。神经网络的知识体现在网络连接的权值上，神经网络的学习主要表现在神经网络权值的逐步计算上。

③ 遗传算法：遗传算法是模拟生物进化过程的算法，它由三个基本过程组成：繁殖（选择）、交叉（重组）、变异（突变）。采用遗传算法可以产生优良的后代，经过若干代的遗传，将得到满足要求的后代即问题得解。

④ 关联规则挖掘算法：关联规则是描述数据之间存在关系的规则，形式为“A1A2…An=>B1B2…Bn”。一般分为两个步骤：求出大数据项集、用大数据项集产生关联规则。

除了上述的常用方法外，还有粗集方法，模糊集合方法，Bayesian Belief Netords，最邻近算法（K-nearest Neighbors Method，kNN）等。

参考答案

（53）C

试题（54）～（56）

部门、员工和项目的关系模式及它们之间的 E-R 图如下所示，其中，关系模式中带实下画线的属性表示主键属性。图中：

部门（<u>部门代码</u>，部门名称，电话）
员工（<u>员工代码</u>，姓名，部门代码，联系方式，薪资）
项目（<u>项目编号</u>，项目名称，承担任务）



若部门和员工关系进行自然连接运算，其结果集为__（54）__元关系。由于员工和项目关系之间的联系类型为__（55）__，所以员工和项目之间的联系需要转换成一个独立的关系模式，该关系模式的主键是__（56）__。

（54）A．5　　B．6　　C．7　　D．8

（55）A．1 对 1　　B．1 对多　　C．多对 1　　D．多对多

（56）A．（项目名称，员工代码）　　B．（项目编号，员工代码）
　　C．（项目名称，部门代码）　　D．（项目名称，承担任务）

试题（54）～（56）分析

本题考查关系数据库 E-R 模型的相关知识。

试题（54）的正确答案是 C。根据题意部门和员工关系进行自然连接运算，应该去掉一个重复属性“部门代码”，所以自然连接运算的结果集为 7 元关系。

试题（55）的正确答案是 D。在 E-R 模型中，用 1　1 表示 1 对 1 联系，用 1　*表示 1 对多联系，用*　*表示多对多联系。

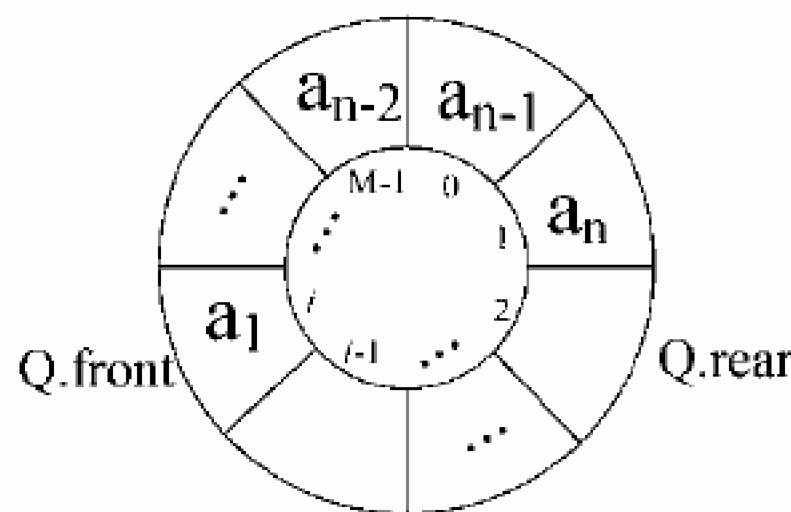
试题（56）的正确答案是 B。因为员工和项目之间是一个多对多的联系，多对多联系的向关系模式转换的规则是：多对多联系只能转换成一个独立的关系模式，关系模式的名称取联系的名称，关系模式的属性取该联系所关联的两个多方实体的主键及联系的属性，关系的码是多方实体的主键构成的属性组。由于员工关系的主键是员工代码，项目关系的主键是项目编号，因此，根据该转换规则试题（56）员工和项目之间的联系的关系模式的主键是（员工代码，项目编号）。

参考答案

（54）C　（55）D　（56）B

试题（57）

设某循环队列 Q 的定义中有 front 和 rear 两个域变量，其中，front 指示队头元素的位置，rear 指示队尾元素之后的位置，如下图所示。若该队列的容量为 M，则其长度为（57）。



（57）A．(Q.rear–Q.front +1)

B．(Q.rear–Q.front+M)

C．(Q.rear–Q.front+1)%M

D．(Q.rear–Q.front+M)%M

试题（57）分析

本题考查数据结构基础知识。

根据图示，可以计算得知(Q.rear–Q.front+M)%M 为队列中的元素个数(即队列长度)。

参考答案

（57）D

试题（58）

设栈 S 和队列 Q 的初始状态为空，元素 a b c d e f g 依次进入栈 S。要求每个元素出栈后立即进入队列 Q，若 7 个元素出队列的顺序为 b d f e c a g，则栈 S 的容量最小应该是（58）。

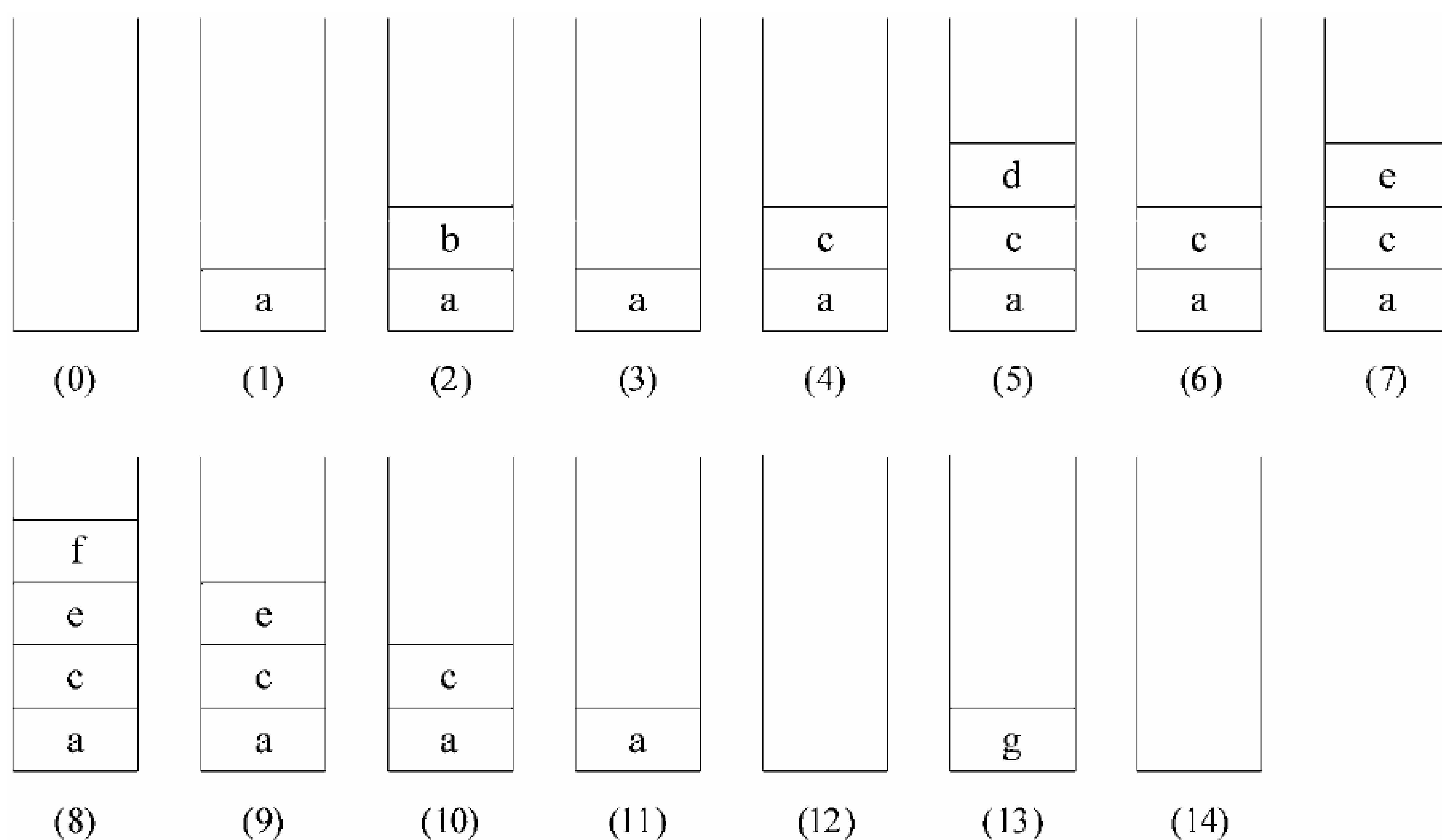
（58）A．5　B．4　C．3　D．2

试题（58）分析

本题考查数据结构基础知识。

根据队列的特点，元素出队的顺序与入队的顺序相同，因此，可知这 7 个元素的出栈顺序为 b d f e c a g。对于入栈序列 a b c d e f g，得到出栈序列 b d f e c a g 的操作过程为：push（a 入）、push（b 入）、pop（b 出）、push（c 入）、push（d 入）、pop（d 出）、

push（e 入）、push（f 入）、pop（f 出）、pop（e 出）、pop（c 出）、pop（a 出）、push（g 入）、pop（g 出），如下图所示，从中可知栈 S 中元素最多时为 4。因此，S 的容量最小为 4。



参考答案

（58）B

试题（59）

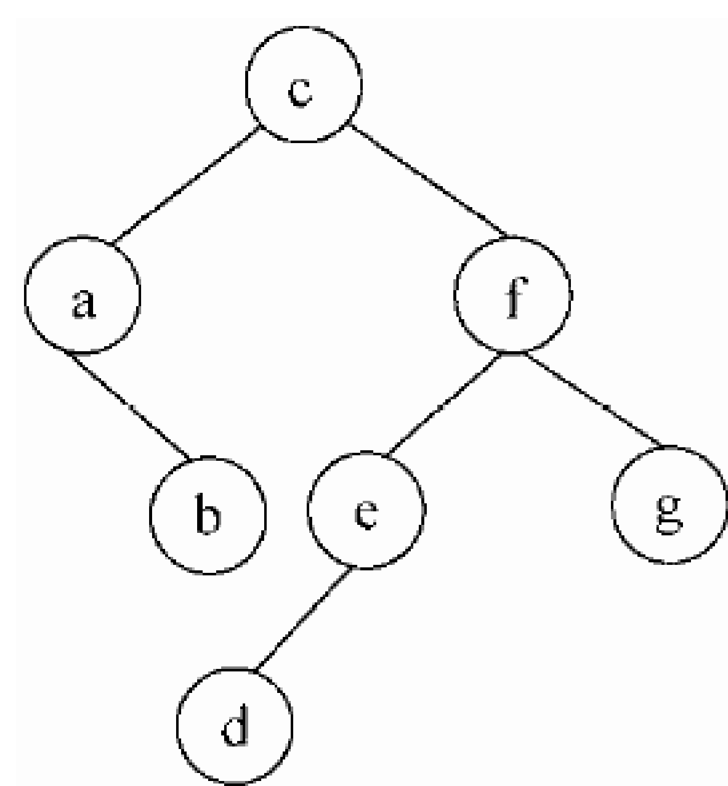
某二叉树的先序遍历序列为 c a b f e d g，中序遍历序列为 a b c d e f g，则该二叉树是__（59）__。

（59）A．完全二叉树　　B．最优二叉树　　C．平衡二叉树　　D．满二叉树

试题（59）分析

本题考查数据结构基础知识。

根据题中所给的遍历序列，可知其对应的二叉树如下图所示。



参考答案

（59）C

试题（60）

对某有序顺序表进行折半查找时，___(60)___不可能构成查找过程中关键字的比较序列。

（60）A．45，10，30，18，25　　B．45，30，18，25，10

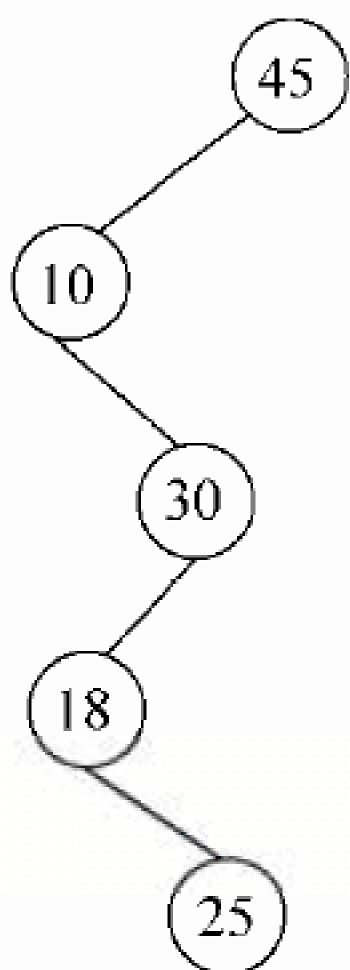
C．10，45，18，30，25　　D．10，18，25，30，45

试题（60）分析

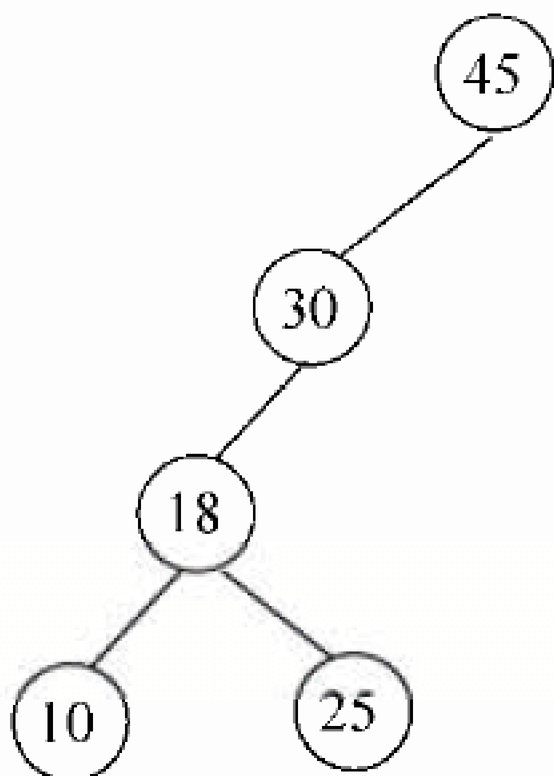
本题考查数据结构基础知识。

进行折半查找时，首先与表中间位置上的元素进行比较，若待查找的元素大于中间元素，则接下来在后半区（是比中间元素更大者组成的有序子表）进行折半查找，否则在前半区（是比中间元素更小者组成的有序子表）进行折半查找。二分查找过程可用二分查找判定树来描述，即大于中间元素时走右分支，小于中间元素时走左分支，等于时查找成功结束。

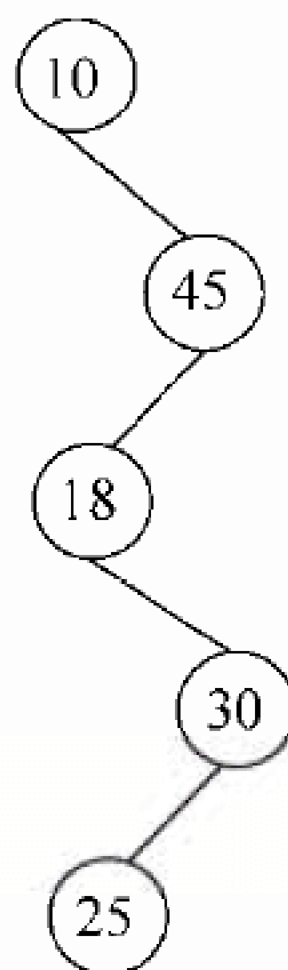
四个选项的比较序列在二分查找判定树上的查找路径表示为：



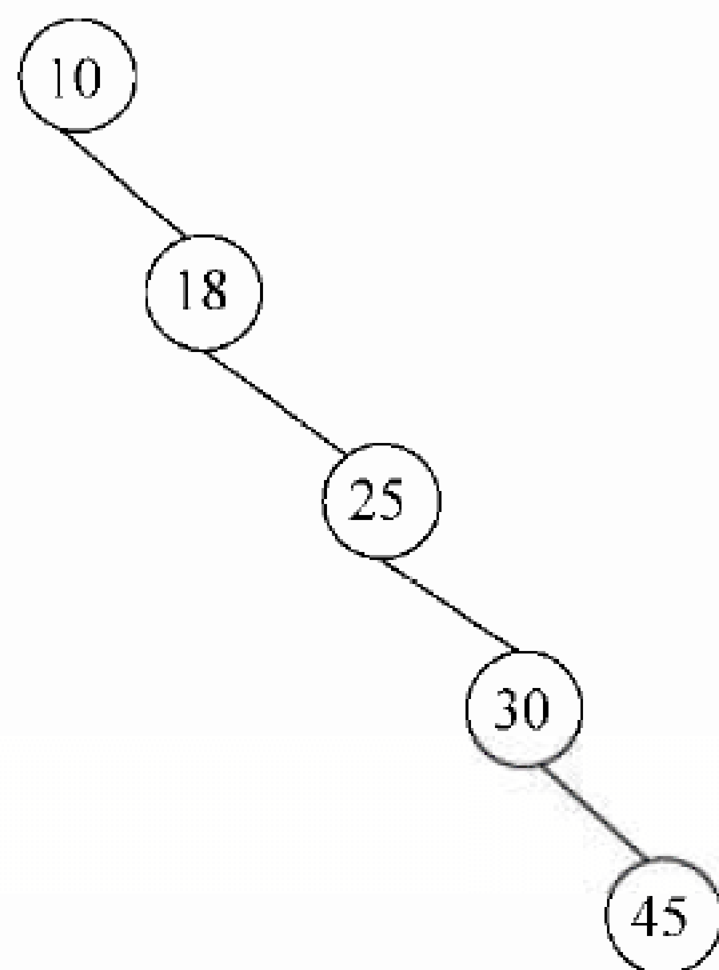
选项 A



选项 B



选项 C



选项 D

显然，选项 B 是不可能的查找路径。

参考答案

（60）B

试题（61）

用某排序方法对一元素序列进行非递减排序时，若该方法可保证在排序前后排序码相同者的相对位置不变，则称该排序方法是稳定的。简单选择排序法排序方法是不稳定的，___(61)___可以说明这个性质。

（61）A．21　48　21*　63　17　　B．17　21　21*　48　63

C．63　21　48　21*　17　　D．21*　17　48　63　21

试题（61）分析

本题考查数据结构基础知识。

对选项 A 进行简单选择排序时，第一趟需交换 17 和 21，导致 21 与 21*的相对位置发生变化，最后的非递减序列为 17 21* 21 48 63，说明简单选择排序是不稳定的排序方法。

参考答案

（61）A

试题（62）、（63）

优先队列通常采用__（62）__数据结构实现，向优先队列中插入一个元素的时间复杂度为__（63）__。

（62）A. 堆　　B. 栈　　C. 队列　　D. 线性表

（63）A. Θ(n)　　B. Θ(1)　　C. Θ(lgn)　　D. $\Theta(n^2)$

试题（62）、（63）分析

本题考查算法设计与分析的相关知识。

优先队列是一种常用的数据结构，通常用堆实现。对应于大顶堆和小顶堆，存在最大优先队列和最小优先队列。以最大优先队列为例，优先队列除了具有堆上的一些操作，如调整堆、构建堆之外，还有获得优先队列的最大元素，抽取出优先队列的最大元素，向优先队列插入一个元素和增大优先队列中某个元素的值。其中，除了获得优先队列的最大元素的时间复杂度为 Θ(1)之外，其他几个操作的时间复杂度均为二叉树的高度，即 Θ(lgn)。

参考答案

（62）A　（63）C

试题（64）、（65）

在 n 个数的数组中确定其第 i（1≤i≤n）小的数时，可以采用快速排序算法中的划分思想，对 n 个元素划分，先确定第 k 小的数，根据 i 和 k 的大小关系，进一步处理，最终得到第 i 小的数。划分过程中，最佳的基准元素选择的方法是选择待划分数组的__（64）__元素。此时，算法在最坏情况下的时间复杂度为（不考虑所有元素均相等的情况）__（65）__。

（64）A. 第一个　　B. 最后一个　　C. 中位数　　D. 随机一个

（65）A. Θ(n)　　B. Θ(lgn)　　C. Θ(nlgn)　　D. $\Theta(n^2)$

试题（64）、（65）分析

本题考查算法设计与分析的相关知识。

中位数的含义：将一组数据按照由小到大（或由大到小）的顺序排列，如果数据的个数是奇数，则处于中间位置的数就是这组数据的中位数；如果数据的个数是偶数，则中间两个数据的平均数就是这组数据的中位数。根据题干的描述，选择的基准元素将数组分得越均匀越好，因此中位数是最佳选择。

对于该问题，若每次都是选择中位数作为基准元素，则时间复杂度的递归式为：

$$T(n) = T(n/2) + cn$$

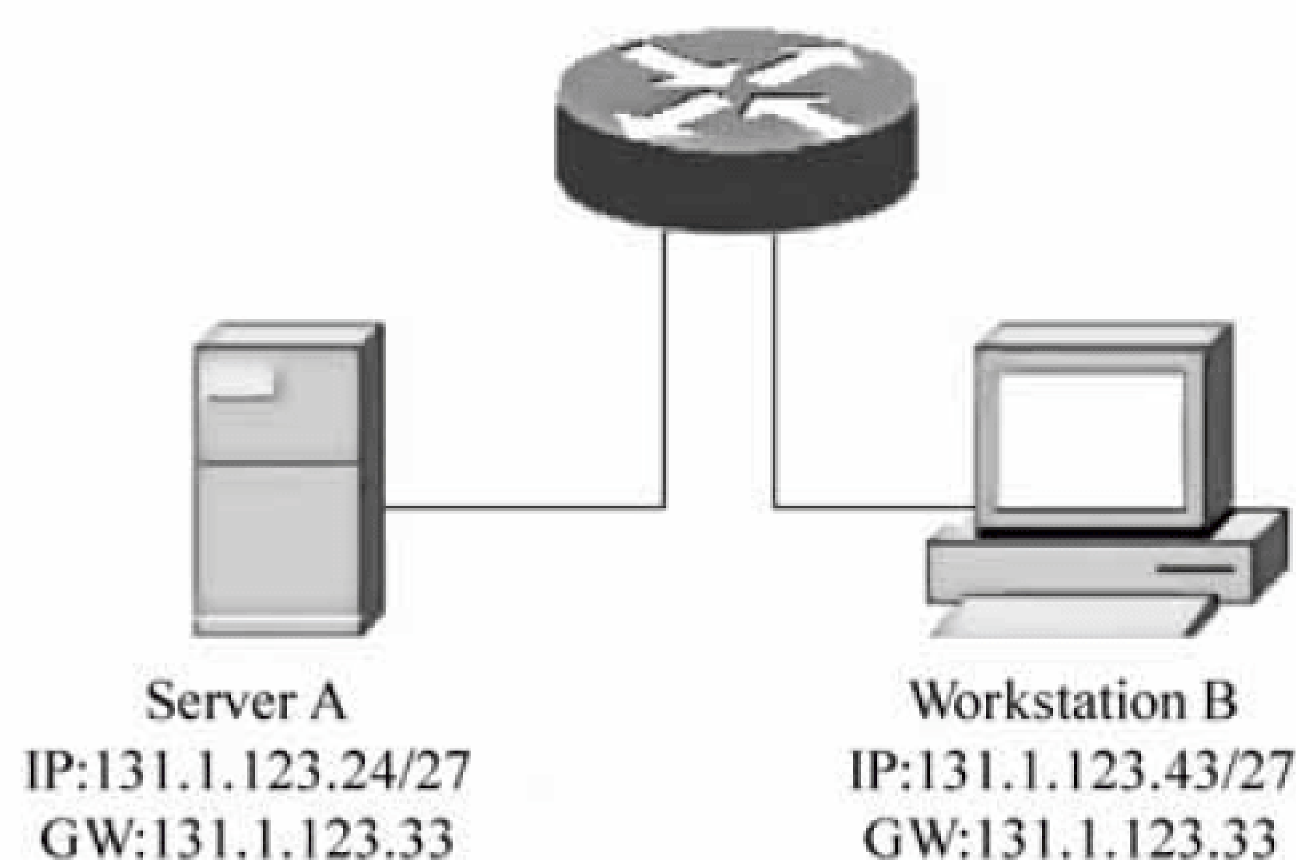
求解该递归式，得到 $T(n) = \Theta(n)$。

参考答案

（64）C　　（65）A

试题（66）

在下图所示的网络配置中，发现工作站 B 无法与服务器 A 通信。__（66）__故障影响了两者互通。



（66）A．服务器 A 的 IP 地址是广播地址
　　　B．工作站 B 的 IP 地址是网络地址
　　　C．工作站 B 与网关不属于同一子网
　　　D．服务器 A 与网关不属于同一子网

试题（66）分析

服务器 A 的 IP 地址 131.1.123.24/27：**10000011.00000001. 01111011.000**11000 服务器 A 的地址不是广播地址。

服务器 A 的网关地址 131.1.123.33：**10000011.00000001. 01111011.001**00001 这个地址与服务器 A 的地址不属于同一个子网。

工作站 B 的 IP 地址 131.1.123.43/27：**10000011.00000001. 01111011.001**01011 这个地址不是网络地址。

工作站 B 的网关地址 131.1.123.33：**10000011.00000001. 01111011.001**00001 工作站 B 与网关属于同一个子网。

参考答案

（66）D

试题（67）

以下关于 VLAN 的叙述中，属于其优点的是__（67）__。

（67）A．允许逻辑地划分网段　　　B．减少了冲突域的数量
　　　C．增加了冲突域的大小　　　D．减少了广播域的数量

试题（67）分析

把局域网划分成多个不同的 VLAN，使得网络接入不再局限于物理位置的约束，这样就简化了在网络中增加、移除和移动主机的操作，特别是动态配置的 VLAN，无论主机在哪里，它都处于自己的 VLAN 中。VLAN 内部可以相互通信，VLAN 之间不能直接通信，必须经过特殊设置的路由器才可以连通。这样做的结果是，通过在较大的局域网中创建不同的 VLAN，可以抵御广播风暴的影响，也可以通过设置防火墙来提高网络的安全性。VLAN 并不能直接增强网络的安全性。

参考答案

（67）A

试题（68）

以下关于 URL 的叙述中，不正确的是__（68）__。

（68）A．使用 www.abc.com 和 abc.com 打开的是同一页面

B．在地址栏中输入 www.abc.com 默认使用 http 协议

C．www.abc.com 中的“www”是主机名

D．www.abc.com 中的“abc.com”是域名

试题（68）分析

本题考查 URL 的使用和格式的基本知识。

URL 由三部分组成：资源类型、存放资源的主机域名、资源文件名。

URL 的一般语法格式为（带方括号[]的为可选项）：

```
protocol :// hostname[:port] / path /filename
```

其中，protocol 指定使用的传输协议，最常见的是 HTTP 或者 HTTPS 协议，也可以有其他协议，如 file、ftp、gopher、mms、ed2k 等；hostname 是指主机名，即存放资源的服务域名或者 IP 地址；port 是指各种传输协议所使用的默认端口号，该选项是可选选项，例如 http 的默认端口号为 80，一般可以省略，如果为了安全考虑，可以更改默认的端口号，这时，该选项是必选的；path 是指路径，有一个或者多个“/”分隔，一般用来表示主机上的一个目录或者文件地址；filename 是指文件名，该选项用于指定需要打开的文件名称。

一般情况下，一个 URL 可以采用“主机名.域名”的形式打开指定页面，也可以单独使用“域名”来打开指定页面，但是这样实现的前提是需进行相应的设置和对应。

参考答案

（68）A

试题（69）、（70）

DHCP 协议的功能是__（69）__；FTP 使用的传输层协议为__（70）__。

（69）A．WINS 名字解析　　B．静态地址分配

C．DNS 名字登录　　D．自动分配 IP 地址

（70）A．TCP　　B．IP　　C．UDP　　D．HDLC

试题（69）、（70）分析

本题考查 DHCP 和 FTP 两个应用协议。

DHCP 协议的功能是自动分配 IP 地址；FTP 协议的作用是文件传输，使用的传输层协议为 TCP。

参考答案

（69）D　　（70）A

试题（71）～（75）

Why Have Formal Documents?

First, writing the decisions down is essential. Only when one writes do the gaps appear and the ＿（71）＿ protrude(突出). The act of writing turns out to require hundreds of mini-decisions, and it is the existence of these that distinguishes clear, exact policies from fuzzy ones.

Second, the documents will communicate the decisions to others. The manager will be continually amazed that policies he took for common knowledge are totally unknown by some member of his team. Since his fundamental job is to keep everybody going in the ＿（72）＿ direction, his chief daily task will be communication, not decision-making, and his documents will immensely ＿（73）＿ this load.

Finally, a manager's documents give him a data base and checklist. By reviewing them ＿（74）＿ he sees where he is, and he sees what changes of emphasis or shifts in direction are neededD．

The task of the manager is to develop a plan and then to realize it. But only the written plan is precise and communicable. Such a plan consists of documents on what, when, how much, where, and who. This small set of critical documents ＿（75）＿ much of the manager's work. If their comprehensive and critical nature is recognized in the beginning, the manager can approach them as friendly tools rather than annoying busywork. He will set his direction much more crisply and quickly by doing so.

（71）A．inconsistencies　B．consistencies　C．steadiness　D．adaptability

（72）A．other　B．different　C．another　D．same

（73）A．extend　B．broaden　C．lighten　D．release

（74）A．periodically　B．occasionally　C．infrequently　D．rarely

（75）A．decides　B．encapsulates　C．realizes　D．recognizes

参考译文

为什么要有正式的文档？

首先，书面记录决策是必要的。只有记录下来，分歧才会明朗，矛盾才会突出。书写这项活动需要上百次的细小决定，正是由于它们的存在，人们才能从令人迷惑的现象中得到清晰、确定的策略。

第二，文档能够作为同其他人的沟通渠道。项目经理常常会不断发现，许多理应被普遍认同的策略，完全不为团队的一些成员所知。正因为项目经理的基本职责是使每个人都向着相同的方向前进，所以他的主要工作是沟通，而不是做出决定。这些文档能极大地减轻他的负担。

最后，项目经理的文档可以作为数据基础和检查列表。通过周期性的回顾，他能清楚项目所处的状态，以及哪些需要重点进行更改和调整。

项目经理的任务是制订计划，并根据计划实现。但是只有书面计划是精确和可以沟通的。计划中包括了时间、地点、人物、做什么、资金。这些少量的关键文档封装了一些项目经理的工作。如果一开始就认识到它们的普遍性和重要性，那么就可以将文档作为工具友好地利用起来，而不会让它成为令人厌烦的繁重任务。通过遵循文档开展工作，项目经理能更清晰和快速地设定自己的方向。

参考答案

（71）A （72）D （73）C （74）A （75）B

第 26 章　2015 上半年软件设计师下午试题分析与解答

试题一（共 15 分）

阅读下列说明和图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

【说明】

某大学为进一步推进无纸化考试，欲开发一考试系统。系统管理员能够创建包括专业方向、课程编号、任课教师等相关考试基础信息，教师和学生进行考试相关的工作。系统与考试有关的主要功能如下。

（1）考试设置。教师制定试题（题目和答案），制定考试说明、考试时间和提醒时间等考试信息，录入参加考试的学生信息，并分别进行存储。

（2）显示并接收解答。根据教师设定的考试信息，在考试有效时间内向学生显示考试说明和题目，根据设定的考试提醒时间进行提醒，并接收学生的解答。

（3）处理解答。根据答案对接收到的解答数据进行处理，然后将解答结果进行存储。

（4）生成成绩报告。根据解答结果生成学生个人成绩报告，供学生查看。

（5）生成成绩单。对解答结果进行核算后生成课程成绩单供教师查看。

（6）发送通知。根据成绩报告数据，创建通知数据并将通知发送给学生；根据成绩单数据，创建通知数据并将通知发送给教师。

现采用结构化方法对考试系统进行分析与设计，获得如图 1-1 所示的上下文数据流图和图 1-2 所示的 0 层数据流图。

【问题 1】（2 分）

使用说明中的词语，给出图 1-1 中的实体 E1～E2 的名称。

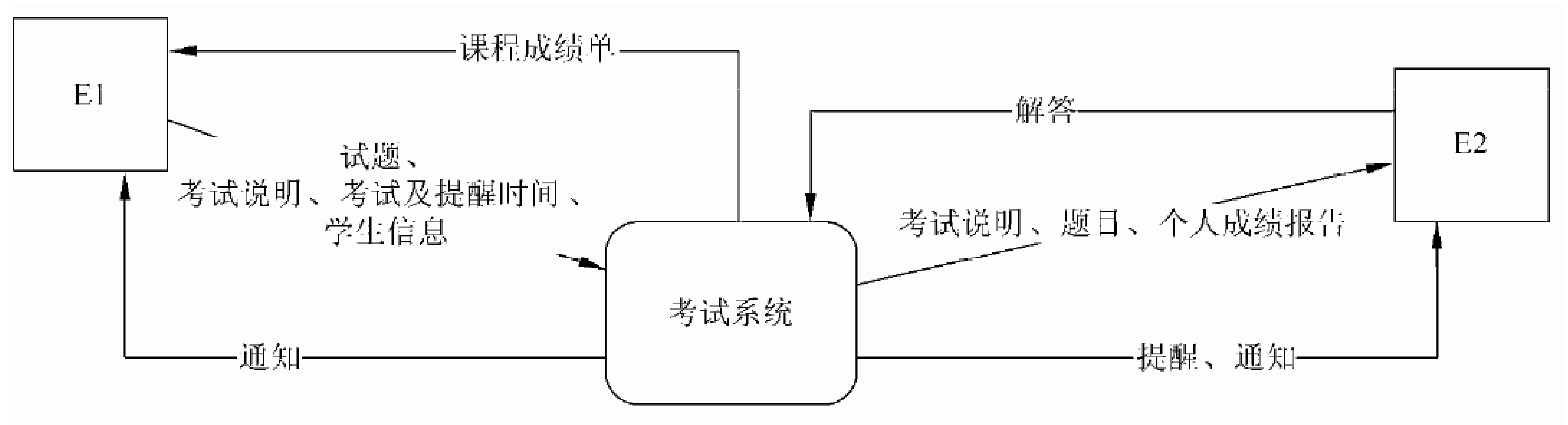


图 1-1　上下文数据流图

【问题 2】（4 分）

使用说明中的词语，给出图 1-2 中的数据存储 D1～D4 的名称。

【问题 3】（4 分）

根据说明和图中词语，补充图 1-2 中缺失的数据流及其起点和终点。

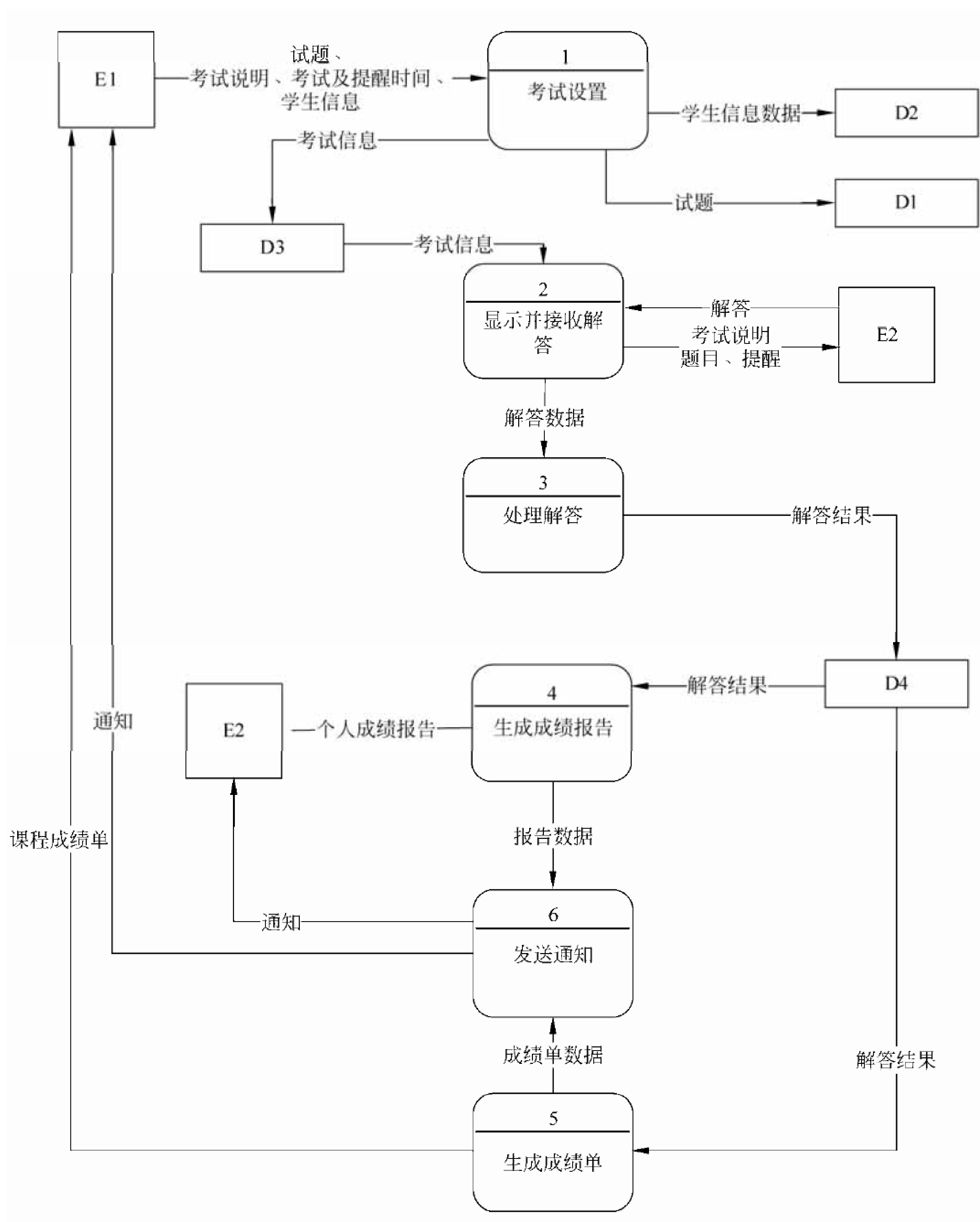


图 1-2　0 层数据流图

【问题 4】（5 分）

在图 1-2 所示的数据流图中，功能（6）发送通知包含创建通知并发送给学生或老师。请分解图 1-2 中加工（6），将分解出的加工和数据流填入答题纸的对应栏内（注：数据流的起点和终点需使用加工的名称描述）。

试题一分析

本题考查采用结构化方法进行系统分析与设计，主要考查数据流图（DFD）的应用，是比较传统的题目，与往年考点类似，要求考生细心分析题目中所描述的内容。

DFD 是一种便于用户理解、分析系统数据流程的图形化建模工具，是系统逻辑模型的重要组成部分。上下文 DFD（顶层 DFD）通常用来确定系统边界，将待开发系统本身看作一个大的加工（处理），然后根据谁为系统提供数据流，谁使用系统提供的数据流，来确定外部实体。建模出的上下文 DFD 中只有唯一的一个加工和一些外部实体，以及这两者之间的输入输出数据流。在上下文确定的系统外部实体以及与外部实体的输入输出数据流的基础上，建模 0 层 DFD，将上下文 DFD 中的加工进一步分解成多个加工，识别这些加工的输入输出数据流，使得所有上下文 DFD 中的输入数据流，经过这些加工之后变换成上下文 DFD 的输出数据流。根据 0 层 DFD 中加工的复杂程度进一步建模加工的内容。

在建模分层 DFD 时，根据需求情况可以将数据存储建模在不同层次的 DFD 中，注意要在绘制下层数据流图时要保持父图与子图平衡。父图中某加工的输入输出数据流必须与它的子图的输入输出数据流在数量和名字上相同，或者父图中的一个输入（或输出）数据流对应于子图中几个输入（或输出）数据流，而子图中组成这些数据流的数据项全体正好是父图中的这一个数据流。

【问题 1】

本问题考查上下文 DFD，要求确定外部实体。考察系统的主要功能，不难发现，针对系统与考试有关的主要功能，涉及到教师和学生，系统管理员不在与考试有关的主要功能中涉及，另外没有提到其他与系统交互的外部实体。根据描述（1）中“教师制定试题等考试信息”等信息，描述（2）中“根据教师设定的考试信息，在考试有效时间内向学生显示考试说明和题目”，从而即可确定 E1 为“教师”实体，E2 为“学生”实体。

【问题 2】

本问题要求确定 0 层数据流图中的数据存储。分析说明中和数据存储有关的描述，说明中（1）中“教师制定试题（题目和答案），制定考试说明、考试时间和提醒时间等考试信息，录入参加考试的学生信息，并分别进行存储”，可知 D1、D2 和 D3 为试题、学生信息和考试信息，再从图 1-2 中流入 D2 的数据流名称“学生信息数据”，确定 D2 是学生信息，流入 D1 的数据流名称为“试题”，确定 D1 为试题，流入 D3 的数据流名称为考试信息，确定 D3 为考试信息。说明中（3）根据答案对接收到的解答数据进行处理，然后将解答结果进行存储，确定 D4 是解答结果。参考其他描述中对数据存储的使

用更多说明，进一步确定 D1～D4 满足上述分析。

【问题 3】

本问题要求补充缺失的数据流及其起点和终点。通过不同层的 DFD 以及说明中描述和图之间的对应关系加以确定。首先对照图 1-1 和图 1-2 的输入、输出数据流，发现数据流的数量和名称均相同，所以，需进一步考查说明中的功能描述和图 1-1 中的数据流的对应关系，以确定缺失的是加工之间还是加工与数据存储之间的数据流。

说明（2）显示并接收解答，需要“根据教师设定的考试信息，在考试有效时间内向学生显示考试说明和题目”，对照图 1-2 可以看出，加工 2 缺少所要显示的题目的输入源，即缺失输入流“题目”，题目存储于数据存储“试题”中，因此，缺少的数据流为从题目（D1）到加工 2 显示并接收解答的题目。说明（3）处理解答，需要“根据答案对接收到的解答数据进行处理”，对照图 1-2 可以看出，加工 3“处理解答”缺少输入流“答案”，而从说明（1）中可以看出“答案”存储在试题（题目和答案）数据存储中（D1），因此确定缺失的一条数据流“答案”，从 D1 或试题到加工 3 或处理解答。

【问题 4】

本问题考查建模分层 DFD 时的分解粒度。在说明（6）发送通知中，“根据成绩报告数据，创建通知数据并将通知发送给学生；根据成绩单数据，创建通知数据并将通知发送给教师。”说明功能（6）发送通知包含创建通知并发送给学生或老师。在图 1-2 中建模为一个加工，完成的功能是依据不同的输入数据流创建通知，然后发送给相应的外部实体老师或学生，因此为了进一步清晰每个加工的职责，需对图 1-2 中原有加工 6 进行分解，分解为“创建通知”和“发送通知”。创建通知针对输入数据流“报告数据”和“成绩单数据”，这两条数据流保持原有的起点，终点即为创建通知。创建通知产生出“通知数据”。“通知数据”作为加工“发送通知”的输入流，进一步根据通知数据是针对哪个外部实体而发送“通知”给相应的学生或者教师。至此，对图 1-2 中原有加工 6 的分解完成。

参考答案

【问题 1】

E1：教师　　E2：学生

【问题 2】

D1：试题（表）或 题目和答案（表）　　D2：学生信息（表）

D3：考试信息（表）　　D4：解答结果（表）

【问题 3】

数据流	起点	终点
答案	D1 或试题（表）或题目和答案（表）	3 或处理解答
题目	D1 或试题（表）或题目和答案（表）	2 或显示并接收解答

【问题 4】

分解为加工：发送通知和加工：创建通知

数　据　流	起　　点	终　　点
报告数据	生成成绩报告	创建通知
成绩单数据	生成成绩单	创建通知
通知数据	创建通知	发送通知

试题二（共 15 分）

阅读下列说明，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某省针对每年举行的足球联赛，拟开发一套信息管理系统，以方便管理球队、球员、主教练、主裁判、比赛等信息。

【需求分析】

（1）系统需要维护球队、球员、主教练、主裁判、比赛等信息。

球队信息主要包括：球队编号、名称、成立时间、人数、主场地址、球队主教练。

球员信息主要包括：姓名、身份证号、出生日期、身高、家庭住址。

主教练信息主要包括：姓名、身份证号、出生日期、资格证书号、级别。

主裁判信息主要包括：姓名、身份证号、出生日期、资格证书号、获取证书时间、级别。

（2）每支球队有一名主教练和若干名球员。一名主教练只能受聘于一支球队，一名球员只能效力于一支球队。每支球队都有自己的唯一主场场地，且场地不能共用。

（3）足球联赛采用主客场循环制，一周进行一轮比赛，一轮的所有比赛同时进行。

（4）一场比赛有两支球队参加，一支球队作为主队身份、另一支作为客队身份参与比赛。一场比赛只能有一名主裁判，每场比赛有唯一的比赛编码，每场比赛都记录比分和日期。

【概念结构设计】

根据需求分析阶段的信息，设计的实体联系图（不完整）如图 2-1 所示。

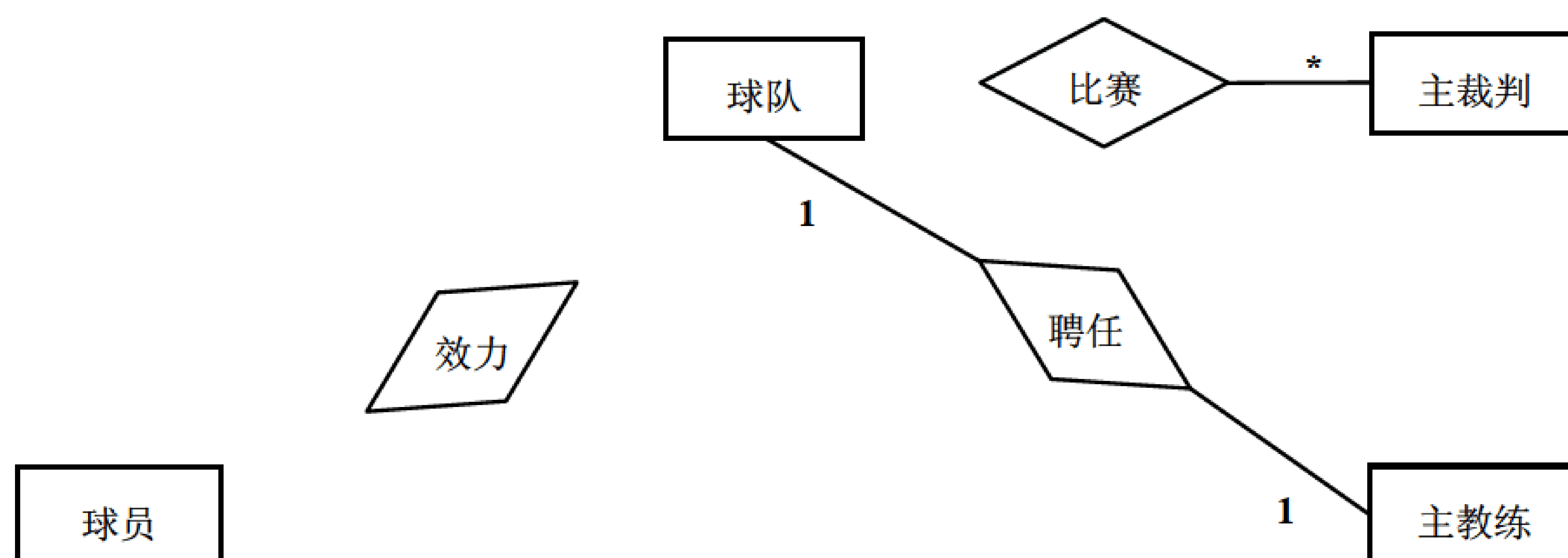


图 2-1　实体联系图

【逻辑结构设计】

根据概念结构设计阶段完成的实体联系图，得出如下关系模式（不完整）：

球队（球队编号，名称，成立时间，人数，主场地址）
球员（姓名，身份证号，出生日期，身高，家庭住址，____（1）____）
主教练（姓名，身份证号，出生日期，资格证书号，级别，____（2）____）
主裁判（姓名，身份证号，出生日期，资格证书号，获取证书时间，级别）
比赛（比赛编码，主队编号，客队编号，主裁判身份证号，比分，日期）

【问题 1】（6 分）

补充图 2-1 中的联系和联系的类型。

图 2-1 中的联系“比赛”应具有的属性是哪些？

【问题 2】（4 分）

根据图 2-1，将逻辑结构设计阶段生成的关系模式中的空（1）、（2）补充完整。

【问题 3】（5 分）

现在系统要增加赞助商信息，赞助商信息主要包括赞助商名称和赞助商编号。

赞助商可以赞助某支球队，一支球队只能有一个赞助商，但赞助商可以赞助多支球队。赞助商也可以单独赞助某些球员，一名球员可以为多个赞助商代言。请根据该要求，对图 2-1 进行修改，画出修改后的实体间联系和联系的类型。

试题二分析

本题考查数据库概念结构设计及向逻辑结构转换的掌握。

此类题目要求考生认真阅读题目，根据题目的需求描述，给出实体间的联系。

【问题 1】

根据题意由“一名球员只能效力于一支球队”可知球队和球员之间为 1:*联系。由“一场比赛有两支球队参加，一支球队作为主队身份、另一支作为客队身份参与比赛”可知球队分别按照“主队”和“客队”两种角色参与“比赛”的*:*联系。“比赛”应具有的属性：比赛编码、比分和日期。

【问题 2】

根据问题 1 分析可知球队和球员之间为 1:*联系，所以在球员关系里应该包括球队的主键，即“球队编号”。根据“每支球队有一名主教练，一名主教练只能受聘于一支球队”可知球队和教练之间为 1:1 联系，而球队关系已经给定，所以需要在主教练关系中包含球队的主键，即“球队编号”。

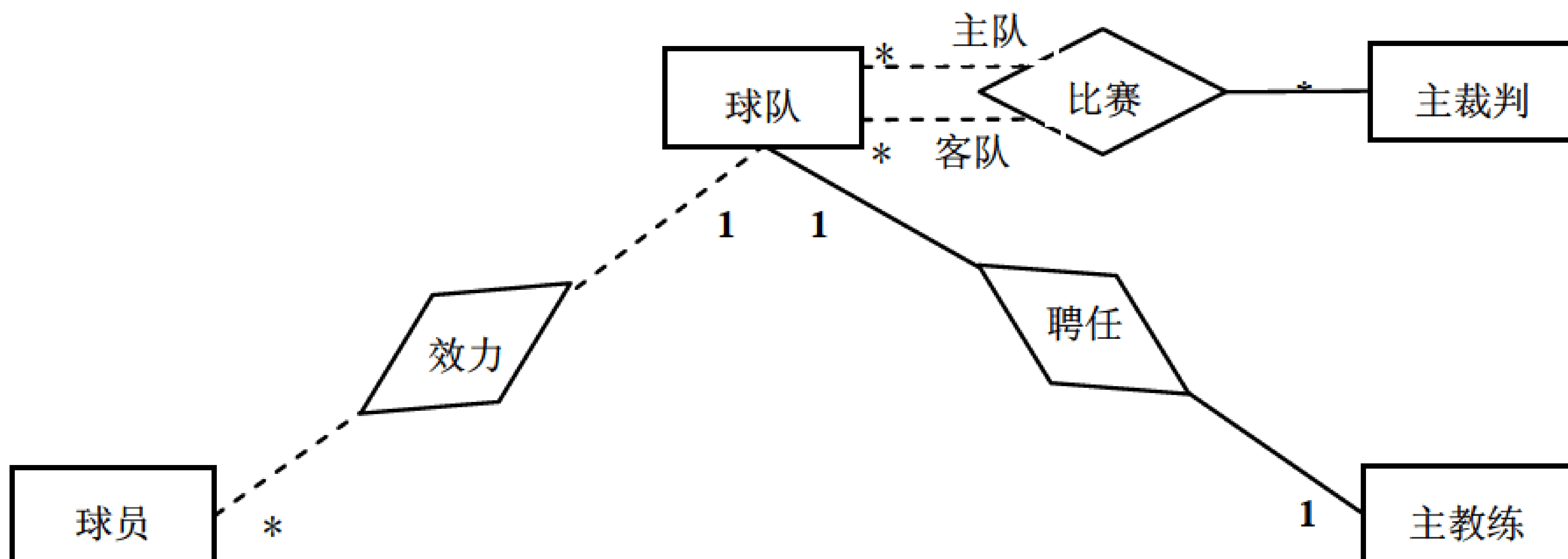
【问题 3】

根据题意由“赞助商可以赞助某支球队，一支球队只能有一个赞助商，但赞助商可以赞助多支球队”可知赞助商和球队之间为 1:*联系。由“赞助商也可以单独赞助某些

球员，一名球员可以为多个赞助商代言”可知赞助商和球员之间为*:*联系。

参考答案

【问题 1】



“比赛”应具有的属性：比赛编码，比分，日期。

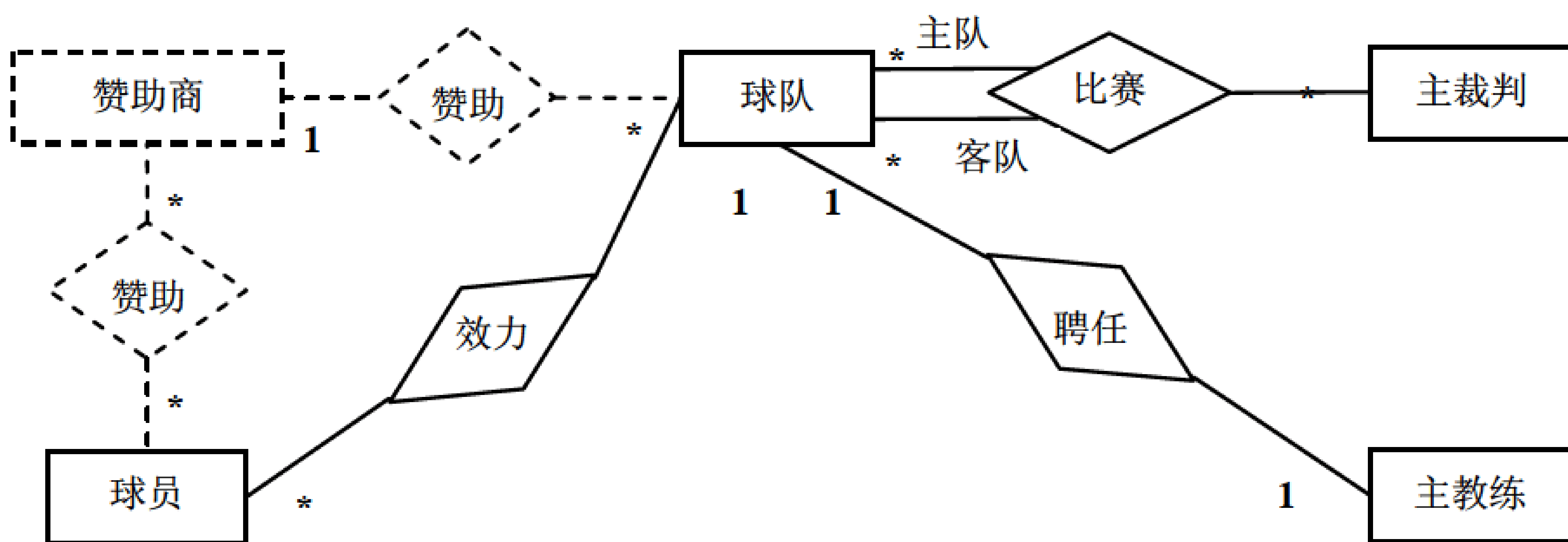
【问题 2】

（1）球队编号

（2）球队编号

【问题 3】

现在系统要增加赞助商信息，赞助商信息主要包括赞助商名称和赞助商编号。



试题三（15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某物品拍卖网站为参与者提供物品拍卖平台，组织拍卖过程，提供在线或线下交易服务。网站主要功能描述如下：

（1）拍卖参与者分为个人参与者和团体参与者两种。不同的团体也可以组成新的团

体参与拍卖活动。网站记录每个参与者的名称。

（2）一次拍卖中，参与者或者是买方，或者是卖方。

（3）一次拍卖只拍出来自一个卖方的一件拍卖品；多个买方可以出价；卖方接受其中一个出价作为成交价，拍卖过程结束。

（4）在拍卖结算阶段，买卖双方可以选择两种成交方式：线下成交，买卖双方在事先约定好的成交地点，当面完成物价款的支付和拍卖品的交付；在线成交，买方通过网上支付平台支付物价款，拍卖品由卖方通过快递邮寄给买方。

一次拍卖过程的基本事件流描述如下：

（1）卖方在网站上发起一次拍卖，并设置本次拍卖的起拍价。

（2）确定拍卖标的以及拍卖标的保留价（若在拍卖时间结束时，所有出价均低于拍卖标的保留价，则本次拍卖失败）。

（3）在网站上发布本次拍卖品的介绍。

（4）买方参与拍卖，给出竞拍价。

（5）卖方选择接受一个竞拍价作为成交价，结束拍卖。

（6）系统记录拍卖成交价，进入拍卖结算阶段。

（7）卖方和买方协商拍卖品成交方式，并完成成交。

现采用面向对象方法对系统进行分析与设计，得到如表3-1所示的类列表以及如图3-1所示的类图，类中关键属性与方法如表3-2所示。

表3-1 物品拍卖网站类列表

序　号	类　名	说　明
C1	SellerRole	一次拍卖中的卖方
C2	Item	拍卖品
C3	Auction	拍卖活动
C4	Sale	拍卖结算
C5	AuctionParticipant	拍卖参与者
C6	Interchange	成交方式
C7	OneParticipant	个人参与者
C8	OfflinePay	线下成交
C9	CompositeParticipant	团体参与者
C10	OnlinePay	在线成交
C11	Bid	拍卖标的
C12	BuyerRole	一次拍卖中的买方

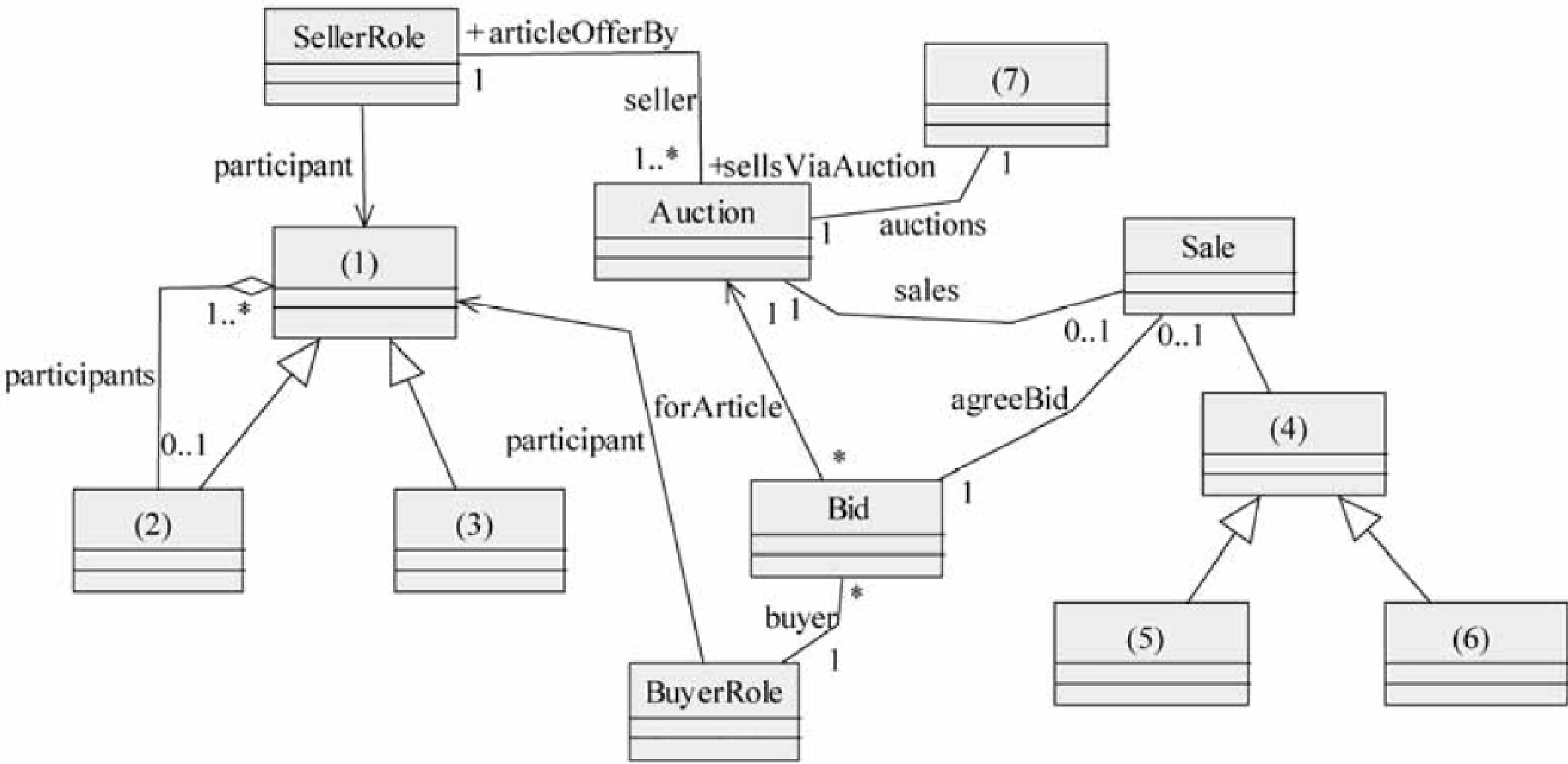


图 3-1　类图

表 3-2　关键属性与方法列表

序　　号	名　　称	说　　明
M1	Name	属性名，用户名称
M2	Description	属性名，拍卖品描述
M3	minBidPrice	属性名，拍卖的起拍价
M4	agreePrice	属性名，拍卖成交价
M5	bidPrice	属性名，拍卖标的保留价
M6	Address	属性名，线下成交地点
M7	sellerAccount	属性名，卖方网上支付账户名
M8	buyerAddress	属性名，买方邮寄地址
M9	placeBidForAuction	方法名，为拍卖品出竞拍价
M10	sellNewArticle	方法名，发起一次拍卖

【问题 1】（7 分）

根据说明中的描述，给出图 3-1 中（1）～（7）所对应的类名（类名使用表 3-1 中给出的序号）。

【问题 2】（5 分）

根据说明中的描述，确定表 3-2 中的属性/方法分别属于哪个类（类名、方法/属性名使用表 3-1、表 3-2 中给出的序号）。

【问题 3】（3 分）

在图 3-1 采用了何种设计模式？以 100 字以内文字说明采用这种设计模式的原因。

试题三分析

本题属于经典的考题，主要考查面向对象分析方法与设计的基本概念。在建模方面，本题中只涉及到了 UML 类图。类图上的考点也是比较常规的对类的识别以及类中属性及方法的确定，题目难度不大。

【问题 1】

图 3-1 共需要确定 7 个类，可以先从图中几个特殊关系处入手，即（1）～（3）和（4）～（6）。

先来分析（1）～（3），这是一个继承+聚集的结构，而且联系的名称“participants”是一个比较明显的提示，说明这个层次结构是与【说明】中的功能描述（1）相对应的。参考表 3-1，与之相关的类是 C5（AuctionParticipant）、C7（OneParticipant）和 C9（CompositeParticipant）。C7、C9 是特殊的参与者，所以（1）处应该为 C5；（2）处应该为 C9，这个聚集关系针对着【说明】中的“不同的团体也可以组成新的团体参与拍卖活动”需求；（3）处为 C7。

结合【说明】和表 3-1，另外一组具有“一般-特殊”关系的类只有 C6（Interchange）、C8（OfflinePay）和 C10（OnlinePay）。显而易见，C8 和 C10 是 C6 的两种具体方式，所以（4）处应该为 C6，（5）、（6）处分别为 C8 和 C10。

这样（7）处对应的类只能是 Item 了。结合【说明】和表 3-1 可知，（7）处对应的类表达的应该是拍卖中的拍卖品，所以（7）处应该是 C2。

【问题 2】

在确定了所有的类之后，确定每个类的属性和方法就比较容易了。完成本问题需要结合【说明】部分中所给出的拍卖过程的基本事件流描述。表 3-2 中的属性/方法与类之间的对应关系如下表所示。

属性/方法序号	所属类的序号	属性/方法序号	所属类的序号
M1	C5	M6	C8
M2	C2	M7	C10
M3	C3	M8	C10
M4	C4	M9	C12
M5	C11	M10	C1

【问题 3】

在【说明】部分有一个很明显的提示：“拍卖参与者分为个人参与者和团体参与者两种。不同的团体也可以组成新的团体参与拍卖活动”。这里很清晰地表达了一种“部分-整体”的层次关系，这种关系非常适合于采用 Composite（组合）设计模式来表达。

Composite 设计模式将对象组合成树形结构以表示“部分-整体”的层次结构。Composite 使得用户对单个对象和组合对象的使用具有一致性。

参考答案

【问题 1】

	类		类
（1）	C5（AuctionParticipant）	（5）	C8（OfflinePay）
（2）	C9（CompositeParticipant）	（6）	C10（OnlinePay）
（3）	C7（OneParticipant）	（7）	C2（Item）
（4）	C6（Interchange）		

注：（5）和（6）的类名可互换。

【问题 2】

属性/方法序号	所属类的序号	属性/方法序号	所属类的序号
M1	C5	M6	C8
M2	C2	M7	C10
M3	C3	M8	C10
M4	C4	M9	C12
M5	C11	M10	C1

【问题 3】

在图 3-1 中使用了 Composite 模式。

以树形结构表示个人参与者和团体参与者之间的“部分-整体”关系，使得对单个对象和组合对象的使用具有一致性。

试题四（共 15 分）

阅读下列说明和 C 代码，回答问题 1 至问题 3，将解答写在答题纸的对应栏内。

【说明】

n-皇后问题是在 n 行 n 列的棋盘上放置 n 个皇后，使得皇后彼此之间不受攻击，其规则是任意两个皇后不在同一行、同一列和相同的对角线上。

拟采用以下思路解决 n-皇后问题：第 i 个皇后放在第 i 行。从第一个皇后开始，对每个皇后，从其对应行（第 i 个皇后对应第 i 行）的第一列开始尝试放置，若可以放置，确定该位置，考虑下一个皇后；若与之前的皇后冲突，则考虑下一列；若超出最后一列，则重新确定上一个皇后的位置。重复该过程，直到找到所有的放置方案。

【C 代码】

下面是算法的 C 语言实现。

（1）常量和变量说明

pos：一维数组，pos[i]表示第 i 个皇后放置在第 i 行的具体位置

count：统计放置方案数

i, j, k：变量

N：皇后数

（2）C 程序

```
#include <stdio.h>
#include <math.h>
#define N 4
/*判断第 k 个皇后目前放置位置是否与前面的皇后冲突*/
int isplace(int pos[], int k){
 int i;
    for(i = 1; i < k; i++){
        if (____(1)____ || fabs(i - k) == fabs(pos[i] - pos[k])) {
            return 0;
        }
    }
    return 1;
}
int main(){
int i, j, count = 1;
int pos[N + 1];
//初始化位置
for (i = 1; i <= N;i++) {
        pos[i] = 0;
}
___(2)___;
while(j >= 1){
        pos[j] = pos[j] + 1;
        /*尝试摆放第 j 个皇后*/
        while(pos[j] <= N && ____(3)____){
            pos[j] = pos[j] + 1;
        }
/*得到一个摆放方案*/
        if (pos[j] <= N && j == N){
            printf("方案%d: ",count++);
            for (i = 1; i <= N;i++){
                printf("%d ", pos[i]);
            }
            printf("\n");
        }
```

```
            /*考虑下一个皇后*/
            if (pos[j] <= N && ____(4)____){
                j = j + 1;
            } else{  // 返回考虑上一个皇后
                pos[j] = 0;
                ____(5)____;
            }
        }
        return 1;
    }
```

【问题 1】（10 分）

根据以上说明和 C 代码，填充 C 代码中的空（1）～（5）。

【问题 2】（2 分）

根据以上说明和 C 代码，算法采用了____(6)____设计策略。

【问题 3】（3 分）

上述 C 代码的输出为：

____(7)____

试题四分析

本题考查算法设计和 C 程序设计语言的相关知识。

此类题目要求考生认真阅读题目，理解算法思想，并思考将算法思想转化为具体的程序设计语言的代码。

【问题 1】

根据题干描述。空（1）所在的代码行判断皇后合法放置的约束条件，即不在同一行，这通过把第 i 个皇后放在第 i 行实现，条件“fabs(i−k) == fabs(pos[i]−pos[k])”判断的是当前摆放的皇后是否与之前摆放的皇后在同一对角线上。因此，空（1）判断的是当前摆放的皇后是否和之前摆放的皇后在同一列上，即应填入“pos[i] == pos[k]”。

根据算法思想和主函数上下文，空（2）处应该考虑第 1 个皇后，即初始化 j 为 1，空（2）填写“j = 1”。空（3）所在的行是判断放置第 j 个皇后的位置是否合适，“pos[j] <= N”表示在该行的合法列上，但还需要进一步判断是否与前面的皇后有冲突，根据满足条件后的语句，尝试放入下一列，因此空（3）处填入“!isplace(pos, j)”。根据前面的注释，空（4）所在的行是考虑下一个皇后，其条件是，当前皇后找到了合适的位置，而且还存在下一个皇后，因此空（4）处应填入“j < N”。根据下面的注释，若当前皇后没有找到合适的位置，则应回溯，即再次考虑上一个皇后的位置，因此空（5）处填入“j = j − 1”。

【问题 2】

从上述题干的叙述和 C 代码很容易看出，从第一个皇后开始，对每个皇后总是从第一个位置开始尝试，找到可以放置的合法位置；若某个皇后在对应的行上没有合法位置，则回溯到上一个皇后，尝试将上一个皇后放置另外的位置。这是典型的深度优先的系统搜索方式，即回溯法的思想。

【问题 3】

四皇后问题的答案为：

方案 1：　2　4　1　3

方案 2：　3　1　4　2

如表 4-1 所示：

表 4-1

方案 1

方案 2

参考答案

【问题 1】

（1）pos[i] == pos[k]　　　或其等价形式

（2）j = 1

（3）!isplace(pos, j)　　　或其等价形式

（4）j < N　　　或其等价形式

（5）j = j – 1　　　或其等价形式

【问题 2】

（6）回溯

【问题 3】

（7）方案 1：　2　4　1　3

　　方案 2：　3　1　4　2

试题五（15 分）

阅读下列说明和 C++代码，将应填入＿（n）＿处的字句写在答题纸的对应栏内。

【说明】

某图书管理系统中管理着两种类型的文献：图书和论文。现在要求统计所有馆藏

文献的总页码（假设图书馆中有一本 540 页的图书和两篇各 25 页的论文，那么馆藏文献的总页码就是 590 页）。采用 Visitor（访问者）模式实现该要求，得到如图 5-1 所示的类图。

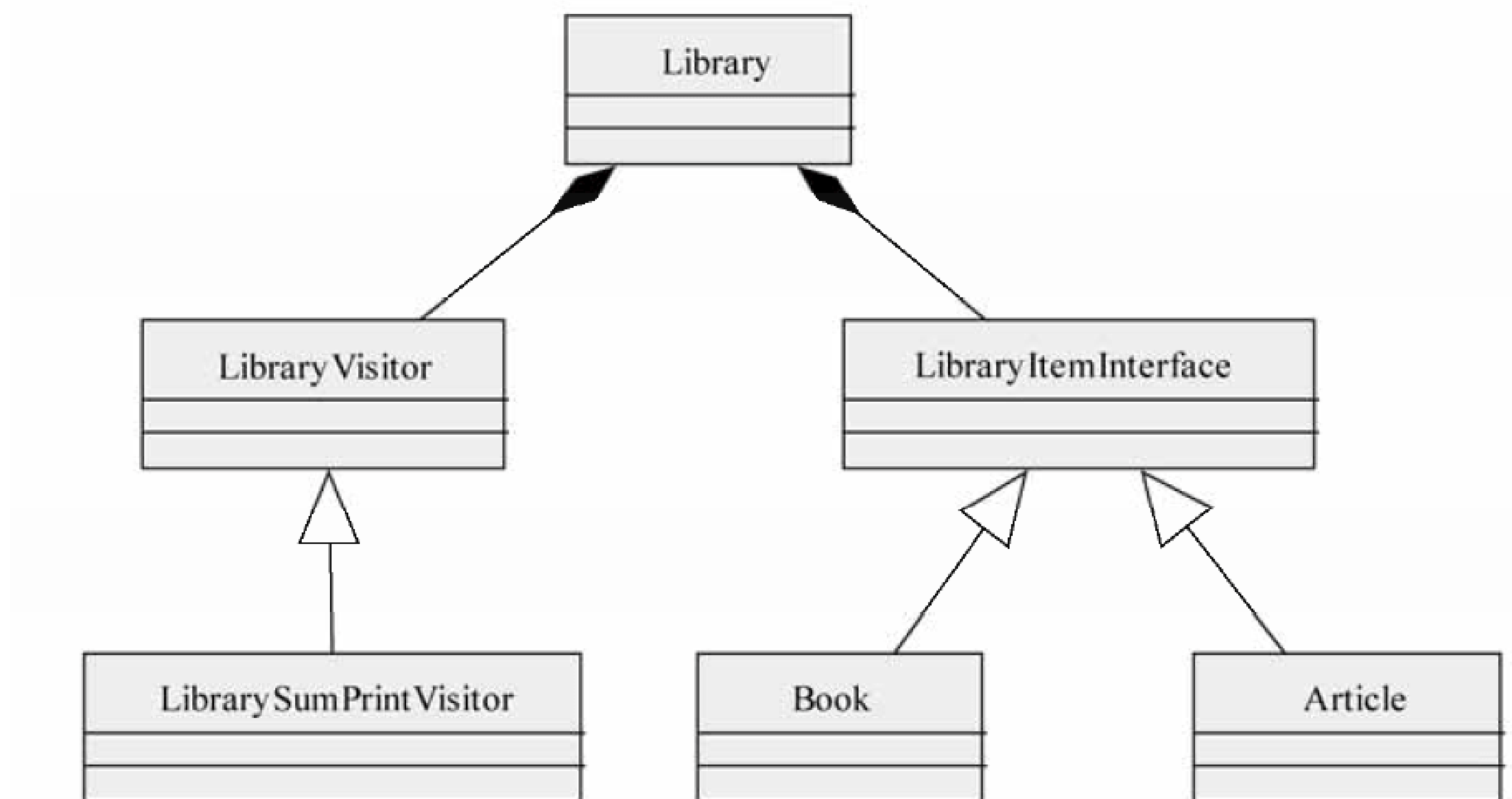


图 5-1　Visitor 模式类图

【C++代码】

```
class LibraryVisitor;
class LibraryItemInterface {
public:
    (1)    ;
};
class Article : public LibraryItemInterface {
private:
    string  m_title;     //论文名
    string  m_author;    //论文作者
    int m_start_page;
    int m_end_page;
public:
    Article(string p_author, string p_title, int p_start_page,int
    p_end_page );
    int getNumberOfPages();
    void accept(LibraryVisitor* visitor);
};
```

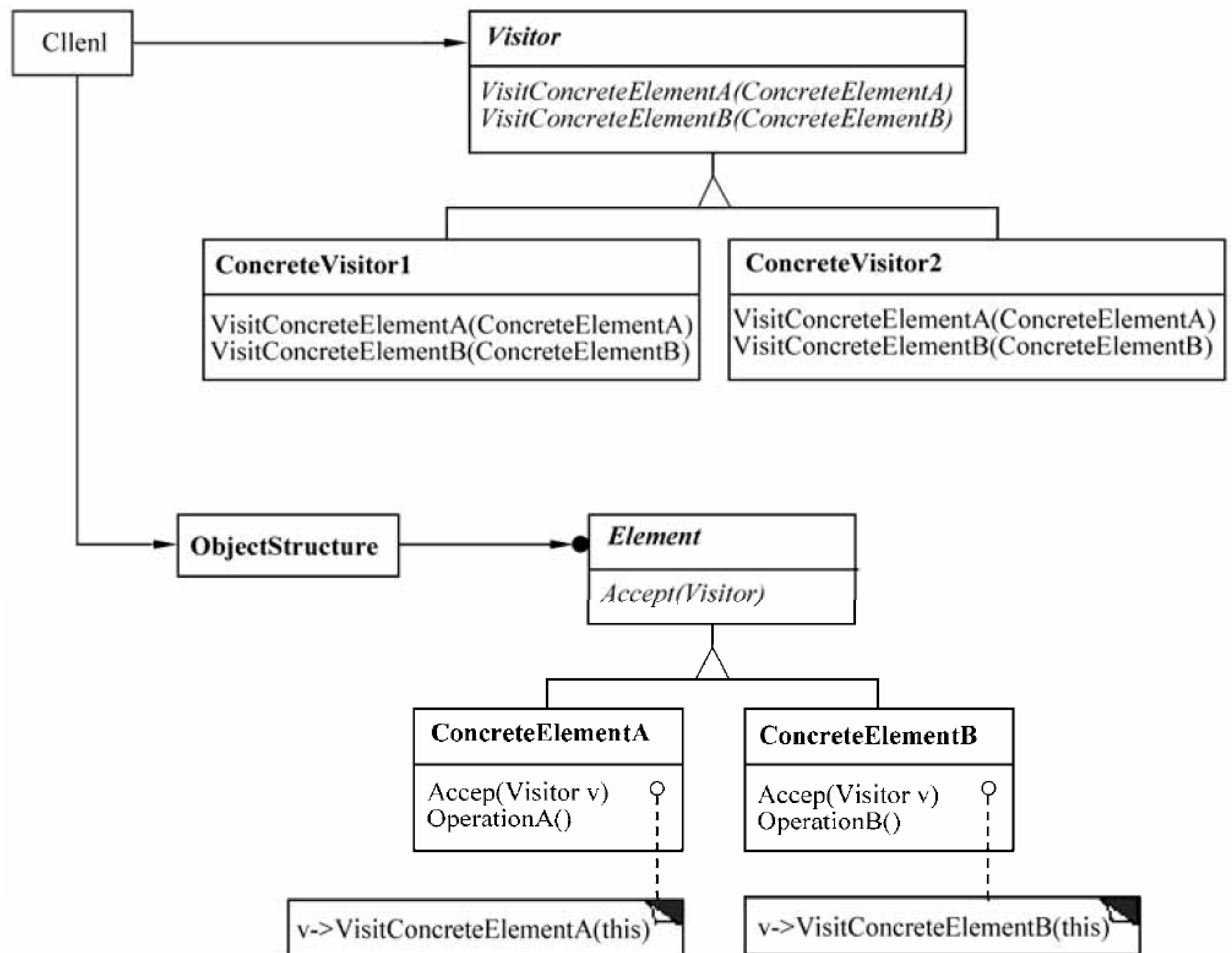
```
class Book : public LibraryItemInterface {
private:
    string  m_title;     //书名
    string  m_author;    //作者
    int m_pages;         //页数
public:
    Book(string p_author, string p_title, int p_pages);
    int getNumberOfPages();
    void accept(LibraryVisitor* visitor);
};
class LibraryVisitor {
public:
    ____(2)____;
    ____(3)____;
    virtual void printSum() = 0;
};
class LibrarySumPrintVisitor : public LibraryVisitor {    // 打印总页数
private:
    int sum;
public:
    LibrarySumPrintVisitor();
    void visit(Book* p_book);
    void visit(Article* p_article);
    void printSum();
};
// visitor.cpp
int Article::getNumberOfPages(){
    return m_end_page - m_start_page;
}
void Article::accept(LibraryVisitor* visitor){ ____(4)____;  }
Book::Book(string p_author, string p_title, int p_pages ){
    m_title = p_title;
    m_author = p_author;
    m_pages = p_pages;
}
int Book::getNumberOfPages(){       return m_pages;  }
void Book::accept(LibraryVisitor* visitor){ ____(5)____;  }
// 其余代码省略
```

试题五分析

本题考查 Visitor（访问者）模式的基本概念和应用。

访问者模式是行为设计模式中的一种。行为模式不仅描述对象或类的模式，还描述它们之间的通信模式。这些模式刻画了在运行时难以跟踪的复杂的控制流。

访问者模式表示一个作用于某对象结构中的各元素的操作。它使在不改变各元素的类的前提下可以定义作用于这些元素的新操作。此模式的结构图如下图所示。



- Visitor（访问者）为该对象结构中 ConcreteElement 的每一个类声明一个 Visit 操作。该操作的名字和特征标识了发送 Visit 请求给该访问者的哪个类。这使得访问者可以确定正被访问元素的具体的类。这样访问者就可以通过该元素的特定接口直接访问它。
- ConcreteVisitor（具体访问者）实现每个有 Visitor 声明的操作，每个操作实现本算法的一部分，而该算法片段乃是对应于结构中对象的类。ConcreteVisitor 为该算法提供了上下文并存储它的局部状态。这一状态常常在遍历该结构的过程中累积结果。

- Element（元素）定义以一个访问者为参数的 Accept 操作。
- ConcreteElement（具体元素）实现以一个访问者为参数的 Accept 操作。
- ObjectStructure（对象结构）能枚举它的元素；可以提供一个高层的接口以允许该访问者访问它的元素；可以是一个组合或者一个集合，如一个列表或一个无序集合。

本题中类 Library 对应着上图中的 Client，LibraryVisitor 对应着 Visitor，LibrarySumPrintVisitor 对应着 ConcreteVisitor。LibraryItemInterface 对应着上图中的元素部分。下面可以结合程序代码来完成程序填空了。

LibraryItemInterface 在本题中充当着 Element 的作用，其中应定义以一个访问者为参数的 Accept 操作。对照其两个子类 Article 和 Book 的代码，可以得知该操作的原型是 void accept(LibraryVisitor* visitor)。由于该操作的具体实现在子类 Article 和 Book 中，所以这里适用于采用 C++中的纯虚拟函数来实现。由此可以得知，（1）处应填写“virtual void accept(LibraryVisitor* visitor) = 0”。

（2）和（3）空与类 LibraryVisitor 有关。由前文分析已知，LibraryVisitor 对应着访问者模式中的 Visitor，其作用是为类 LibrarySumPrintVisitor 声明 Visit 操作。类 LibrarySumPrintVisitor 需要访问两种不同的元素，每种元素应该对应不同的 visit 操作。再结合类 LibrarySumPrintVisitor 的定义部分，可以得知（2）和（3）处应给出分别以 Book 和 Article 为参数的 visit 方法，同样采用纯虚拟函数机制。因此（2）和（3）处分别为“virtual void visit(Book* p_book) = 0”“virtual void visit(Article* p_article) = 0”。

（4）和（5）处考查的是 accept 接口的实现。由访问者模式的结构图可以看出，在 Book 和 Article 中 accept 方法的实现均为 visitor->visit(this)。

参考答案

（1）virtual void accept(LibraryVisitor* visitor) = 0

（2）virtual void visit(Book* p_book) = 0

（3）virtual void visit(Article* p_article) = 0

（4）visitor->visit(this)

（5）visitor->visit(this)

试题六（15 分）

阅读下列说明和 Java 代码，将应填入 （n） 处的字句写在答题纸的对应栏内。

【说明】

某图书管理系统中管理着两种类型的文献：图书和论文。现在要求统计所有馆藏文献的总页码（假设图书馆中有一本 540 页的图书和两篇各 25 页的论文，那么馆藏文献的总页码就是 590 页）。采用 Visitor（访问者）模式实现该要求，得到如图 6-1 所示的

类图。

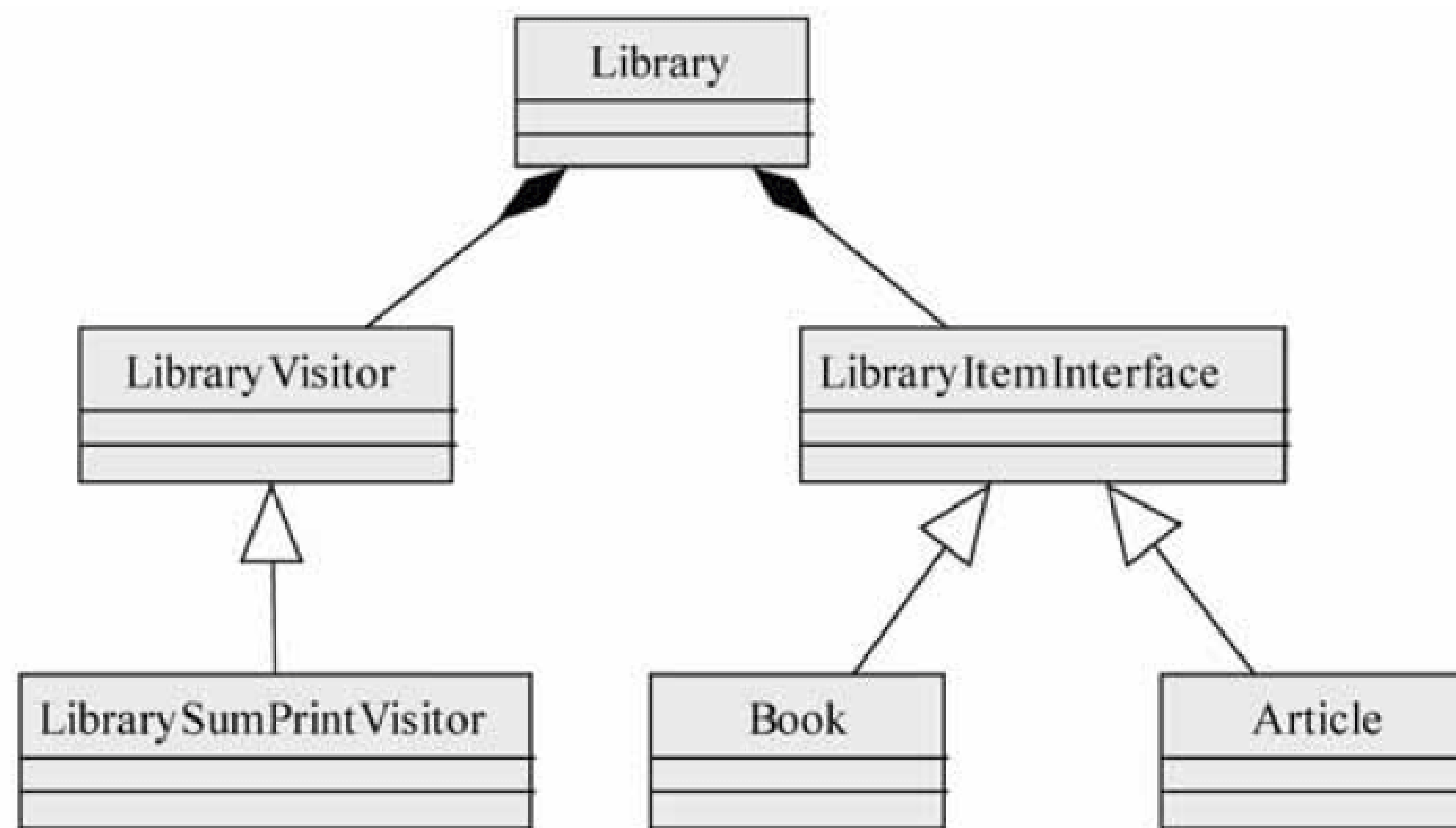


图 6-1　Visitor 模式类图

【Java 代码】

```
import java.util.*;
interface LibraryVisitor {
    ____(1)____;
    ____(2)____;
    void printSum();
}
class LibrarySumPrintVisitor implements LibraryVisitor {    // 打印总页数
    private int sum = 0;
    public void visit(Book p_book){
        sum = sum + p_book.getNumberOfPages();
    }
    public void visit(Article p_article) {
        sum = sum + p_article.getNumberOfPages();
    }
    public void printSum(){
        System.out.println("SUM = " + sum);
    }
}
interface LibraryItemInterface {
    ____(3)____;
}
class Article implements LibraryItemInterface {
    private String  m_title;        //论文名
```

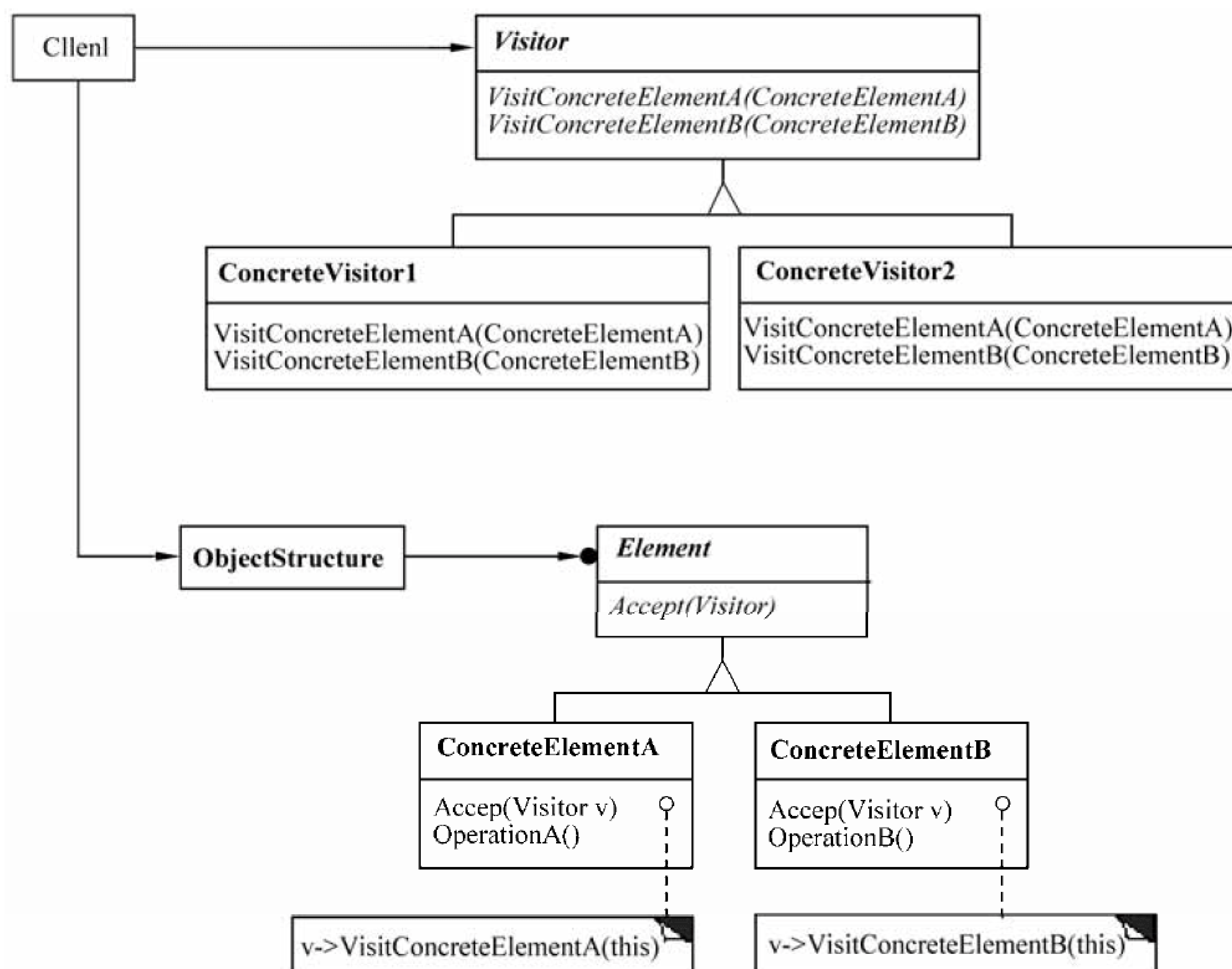
```
        private String  m_author;        //论文作者
        private int m_start_page;
        private int m_end_page;
        public Article(String p_author, String p_title, int p_start_page,int
        p_end_page ){
            m_title = p_title;
            m_author = p_author;
            m_start_page= p_start_page;
            m_end_page= p_end_page;
        }
        public int getNumberOfPages(){
            return m_end_page - m_start_page;
        }
        public void accept(LibraryVisitor visitor){
            ____(4)____;
        }
    }
    class Book implements LibraryItemInterface {
        private String  m_title;         //书名
        private String  m_author;        //书作者
        private int m_pages;             //页数
        public Book(String p_author, String p_title, int p_pages ){
            m_title = p_title;
            m_author = p_author;
            m_pages = p_pages;
        }
        public int getNumberOfPages(){
            return m_pages;
        }
        public void accept(LibraryVisitor visitor){
            ____(5)____;
        }
    }
```

试题六分析

本题考查 Visitor（访问者）模式的基本概念和应用。

访问者模式是行为设计模式中的一种。行为模式不仅描述对象或类的模式，还描述它们之间的通信模式。这些模式刻画了在运行时难以跟踪的复杂的控制流。

访问者模式表示一个作用于某对象结构中的各元素的操作。它使在不改变各元素的类的前提下可以定义作用于这些元素的新操作。此模式的结构图如下图所示。



- Visitor（访问者）为该对象结构中 ConcreteElement 的每一个类声明一个 Visit 操作。该操作的名字和特征标识了发送 Visit 请求给该访问者的哪个类。这使得访问者可以确定正被访问元素的具体的类。这样访问者就可以通过该元素的特定接口直接访问它。
- ConcreteVisitor（具体访问者）实现每个有 Visitor 声明的操作，每个操作实现本算法的一部分，而该算法片段乃是对应于结构中对象的类。ConcreteVisitor 为该算法提供了上下文并存储它的局部状态。这一状态常常在遍历该结构的过程中累积结果。
- Element（元素）定义以一个访问者为参数的 Accept 操作。
- ConcreteElement（具体元素）实现以一个访问者为参数的 Accept 操作。
- ObjectStructure（对象结构）能枚举它的元素；可以提供一个高层的接口以允许该访问者访问它的元素；可以是一个组合或者一个集合，如一个列表或一个无序集合。

本题中类 Library 对应着上图中的 Client，LibraryVisitor 对应着 Visitor，

LibrarySumPrintVisitor 对应着 ConcreteVisitor。LibraryItemInterface 对应着上图中的元素部分。下面可以结合程序代码来完成程序填空了。

（1）和（2）空与类 LibraryVisitor 有关。由前文分析已知，LibraryVisitor 对应着访问者模式中的 Visitor，其作用是为类 LibrarySumPrintVisitor 声明 Visit 操作。类 LibrarySumPrintVisitor 需要访问 2 种不同的元素，每种元素应该对应不同的 visit 操作。再结合类 LibrarySumPrintVisitor 的定义部分，可以得知（2）和（3）处应给出分别以 Book 和 Article 为参数的 visit 方法。因此（1）和（2）处分别为“void visit(Book p_book)”“void visit(Article p_article)”。

LibraryItemInterface 在本题中充当着 Element 的作用，其中应定义以一个访问者为参数的 Accept 操作。对照实现该接口的两个类 Article 和 Book 的代码，可以得知该操作的原型是 void accept(LibraryVisitor visitor)。由此可以得知，（3）处应填写“void accept(LibraryVisitor visitor)”。

（4）和（5）处考查的是 accept 接口的实现。由访问者模式的结构图可以看出，在 Book 和 Article 中 accept 方法的实现均为 visitor.visit(this)。

参考答案

（1）void visit(Book p_book)

（2）void visit(Article p_article)

（3）void accept(LibraryVisitor visitor)

（4）visitor.visit(this)

（5）visitor.visit(this)

第 27 章　2015 下半年软件设计师上午试题分析与解答

试题（1）

CPU 是在__（1）__结束时响应 DMA 请求的。

（1）A．一条指令执行　　B．一段程序
　　C．一个时钟周期　　D．一个总线周期

试题（1）分析

本题考查计算机组成基础知识。

DMA 控制器在需要的时候代替 CPU 作为总线主设备，在不受 CPU 干预的情况下，控制 I/O 设备与系统主存之间的直接数据传输。DMA 操作占用的资源是系统总线，而 CPU 并非在整个指令执行期间即指令周期内都会使用总线，故 DMA 请求的检测点设置在每个机器周期也即总线周期结束时执行，这样使得总线利用率最高。

参考答案

（1）D

试题（2）

虚拟存储体系由__（2）__两级存储器构成。

（2）A．主存-辅存　　B．寄存器-Cache
　　C．寄存器-主存　　D．Cache-主存

试题（2）分析

本题考查计算机组成基础知识。

计算机中不同容量、不同速度、不同访问形式、不同用途的各种存储器形成的是一种层次结构的存储系统。所有的存储器设备按照一定的层次逻辑关系通过软硬件连接起来，并进行有效的管理，就形成了存储体系。不同层次上的存储器发挥着不同的作用。一般计算机系统中主要有两种存储体系：Cache 存储体系由 Cache 和主存储器构成，主要目的是提高存储器速度，对系统程序员以上均透明；虚拟存储体系由主存储器和在线磁盘存储器等辅存构成，主要目的是扩大存储器容量，对应用程序员透明。

参考答案

（2）A

试题（3）

浮点数能够表示的数的范围是由其__（3）__的位数决定的。

（3）A．尾数　　B．阶码　　C．数符　　D．阶符

试题（3）分析

本题考查计算机组成基础知识。

在计算机中使用了类似于十进制科学计数法的方法来表示二进制实数，因其表示不同的数时小数点位置的浮动不固定而取名浮点数表示法。浮点数编码由两部分组成：阶码（即指数，为带符号定点整数，常用移码表示，也有用补码的）和尾数（是定点纯小数，常用补码表示，或原码表示）。因此可以知道，浮点数的精度由尾数的位数决定，表示范围的大小则主要由阶码的位数决定。

参考答案

（3）B

试题（4）

在机器指令的地址字段中，直接指出操作数本身的寻址方式称为 （4） 。

（4）A．隐含寻址　　B．寄存器寻址　　C．立即寻址　　D．直接寻址

试题（4）分析

本题考查计算机组成基础知识。

随着主存增加，指令本身很难保证直接反映操作数的值或其地址，必须通过某种映射方式实现对所需操作数的获取。指令系统中将这种映射方式称为寻址方式，即指令按什么方式寻找（或访问）到所需的操作数或信息（例如转移地址信息等）。可以被指令访问到的数据和信息包括通用寄存器、主存、堆栈及外设端口寄存器等。

指令中地址码字段直接给出操作数本身，而不是其访存地址，不需要访问任何地址的寻址方式被称为立即寻址。

参考答案

（4）C

试题（5）

内存按字节编址从 B3000H 到 DABFFH 的区域其存储容量为 （5） 。

（5）A．123kB　　B．159kB　　C．163kB　　D．194kB

试题（5）分析

本题考查计算机组成基础知识。

直接计算 16 进制地址包含的存储单元个数即可。

DABFFH – B3000H + 1 = 27C00H = 162816 = 159k，按字节编址，故此区域的存储容量为 159kB。

参考答案

（5）B

试题（6）

CISC 是 （6） 的简称。

（6）A．复杂指令集系统计算机　　B．超大规模集成电路

C．精简指令集系统计算机　　D．超长指令字

试题（6）分析

本题考查计算机组成与结构基础知识。

计算机技术发展使得机器性能提高，随着高级语言的发展，程序员需要更强大的命令，指令集往往结合应用需要不断扩展，推动了指令集越来越复杂，形成了 CISC，即 Complex Instruction Set Computer，就是使用复杂指令集系统的计算机。与其对应的是 RISC，即 Reduced Instruction Set Computer，精简指令集系统的计算机。

参考答案

（6）A

试题（7）

__（7）__不属于主动攻击。

（7）A．流量分析　　B．重放　　C．IP 地址欺骗　　D．拒绝服务

试题（7）分析

本题考查的是网络攻击的基础知识。

网络攻击有主动攻击和被动攻击两类。其中主动攻击是指通过一系列的方法，主动向被攻击对象实施破坏的一种攻击方式，例如重放攻击、IP 地址欺骗、拒绝服务攻击等均属于攻击者主动向攻击对象发起破坏性攻击的方式。流量分析攻击是通过持续检测现有网络中的流量变化或者变化趋势，而得到相应信息的一种被动攻击方式。

参考答案

（7）A

试题（8）

防火墙不具备__（8）__功能。

（8）A．记录访问过程　　B．查毒　　C．包过滤　　D．代理

试题（8）分析

本题考查的是防火墙基础知识。

防火墙是一种放置在网络边界上，用于保护内部网络安全的网络设备。它通过对流经数据流进行分析和检查，可实现对数据包的过滤、保存用户访问网络的记录和服务器代理功能。防火墙不具备检查病毒的功能。

参考答案

（8）B

试题（9）

根据下图所示的输出信息，可以确定的是：__（9）__。

（9）A．本地主机正在使用的端口号是公共端口号

B．192.168.0.200 正在与 128.105.129.30 建立连接

C．本地主机与 202.100.112.12 建立了安全连接

D. 本地主机正在与 100.29.200.110 建立连接

```
C:\> netstat -n

Active Connections

  Proto   Local Address          Foreign Address        State
  TCP     192.168.0.200:2011     202.100.112.12:443     ESTABLISHED
  TCP     192.168.0.200:2038     100.29.200.110:110     TIME_WAIT
  TCP     192.168.0.200:2052     128.105.129.30:80      ESTABLISHED
```

试题（9）分析

本题考查网管命令 netstat -n 的含义。

从 netstat -n 的输出信息中可以看出，本地主机 192.168.0.200 使用的端口号 2011、2038、2052 都不是公共端口号。根据状态提示信息，其中已经与主机 128.105.129.30 建立了连接，与主机 100.29.200.110 正在等待建立连接，与主机 202.100.112.12 已经建立了安全连接。

参考答案

（9）C

试题（10）

以下著作权权利中，__（10）__的保护期受时间限制。

（10）A. 署名权　B. 修改权　C. 发表权　D. 保护作品完整权

试题（10）分析

我国著作权法在第 10 条对权利内容作了较为详尽而具体的规定，指明著作权的内容包括人身权利和财产权利。著作人身权是指作者享有的与其作品有关的以人格利益为内容的权利，也称为精神权利，包括发表权、署名权、修改权和保护作品完整权。著作人身权与作者的身份紧密联系，永远属于作者本人，即使作者死亡，其他任何人不能再拥有它。所以，我国著作权法第 20 条规定“作者的署名权、修改权、保护作品完整权的保护期不受限制。”

发表权是属于人身权利，但发表权是一次性权利，即发表权行使一次后，不再享有发表权。发表权是指决定作品是否公之于众的权利，作品一经发表，就处于公知状态，对处于公知状态的作品，作者不再享有发表权，以后再次使用作品与发表权无关，而是行使作品的使用权。

参考答案

（10）C

试题（11）

王某在其公司独立承担了某综合信息管理系统软件的程序设计工作。该系统交付用户、投入试运行后，王某辞职，并带走了该综合信息管理系统的源程序，拒不交还公司。王某认为，综合信息管理系统源程序是他独立完成的，他是综合信息管理系统源程序的

软件著作权人。王某的行为＿（11）＿。

（11）A．侵犯了公司的软件著作权　　B．未侵犯公司的软件著作权

C．侵犯了公司的商业秘密权　　D．不涉及侵犯公司的软件著作权

试题（11）分析

王某的行为侵犯了公司的软件著作权。因为王某作为公司的职员，完成的某一综合信息管理系统软件是针对其本职工作中明确指定的开发目标而开发的软件。该软件应为职务作品，并属于特殊职务作品。公司对该软件享有除署名权外的软件著作权的其他权利，而王某只享有署名权。王某持有该软件源程序不归还公司的行为，妨碍了公司正常行使软件著作权，构成对公司软件著作权的侵犯，应承担停止侵权法律责任，交还软件源程序。

参考答案

（11）A

试题（12）

声音（音频）信号的一个基本参数是频率，它是指声波每秒钟变化的次数，用 Hz 表示。人耳能听到的音频信号的频率范围是＿（12）＿。

（12）A．0Hz ～ 20 kHz　　B．0Hz ～ 200 kHz

C．20Hz ～ 20 kHz　　D．20Hz ～ 200 kHz

试题（12）分析

声音是通过空气传播的一种连续的波，称为声波。声波在时间和幅度上都是连续的模拟信号，通常称为模拟声音（音频）信号。人们对声音的感觉主要有音量、音调和音色。音量又称音强或响度，取决于声音波形的幅度，也就是说，振幅的大小表明声音的响亮程度或强弱程度。音调与声音的频率有关，频率高则声音高昂，频率低则声音低沉。而音色是由混入基音的泛音所决定的，每个基音都有其固有的频率和不同音强的泛音，从而使得声音具有其特殊的音色效果。人耳能听得到的音频信号的频率范围是 20Hz～20kHz，包括：话音（300Hz～3400Hz）、音乐（20Hz～20kHz）、其他声音（如风声、雨声、鸟叫声、汽车鸣笛声等，其带宽范围也是 20Hz～20kHz），频率小于 20Hz 声波信号称为亚音信号（次音信号），高于 20kHz 的信号称为超音频信号（超声波）。

参考答案

（12）C

试题（13）

颜色深度是表达图像中单个像素的颜色或灰度所占的位数（bit）。若每个像素具有 8 位的颜色深度，则可表示＿（13）＿种不同的颜色。

（13）A．8　　B．64　　C．256　　D．512

试题（13）分析

颜色深度是表达图像中单个像素的颜色或灰度所占的位数（bit），它决定了彩色图

像中可出现的最多颜色数，或者灰度图像中的最大灰度等级数。8 位的颜色深度，表示每个像素有 8 位颜色位，可表示 2^8=256 种不同的颜色或灰度等级。表示一个像素颜色的位数越多，它能表达的颜色数或灰度等级就越多，其深度越深。

图像深度是指存储每个像素（颜色或灰度）所用的位数（bit），它也是用来度量图像的分辨率的。像素深度确定彩色图像的每个像素可能有的颜色数，或者确定灰度图像的每个像素可能有的灰度级数。如一幅图像的图像深度为 b 位，则该图像的最多颜色数或灰度级为 2^b 种。显然，表示一个像素颜色的位数越多，它能表达的颜色数或灰度级就越多。例如，只有 1 个分量的单色图像（黑白图像），若每个像素有 8 位，则最大灰度数目为 2^8=256；一幅彩色图像的每个像素用 R、G、B 三个分量表示，若 3 个分量的像素位数分别为 4、4、2，则最大颜色数目为 $2^{4+4+2}=2^{10}$=1024，就是说像素的深度为 10 位，每个像素可以是 2^{10} 种颜色中的一种。本题给出 8 位的颜色深度，则表示该图像具有 2^8=256 种不同的颜色或灰度等级。

参考答案

（13）C

试题（14）

视觉上的颜色可用亮度、色调和饱和度 3 个特征来描述。其中饱和度是指颜色的<u>（14）</u>。

（14）A. 种数　　B. 纯度　　C. 感觉　　D. 存储量

试题（14）分析

饱和度是指颜色的纯度，即颜色的深浅，或者说掺入白光的程度，对于同一色调的彩色光，饱和度越深颜色越纯。当红色加入白光之后冲淡为粉红色，其基本色调仍然是红色，但饱和度降低。也就是说，饱和度与亮度有关，若在饱和的彩色光中增加白光的成分，即增加了光能，而变得更亮了，但是其饱和度却降低了。对于同一色调的彩色光，饱和度越高，颜色越纯。如果在某色调的彩色光中，掺入其他彩色光，将引起色调的变化，而改变白光的成分只引起饱和度的变化。高饱和度的深色光可掺入白色光被冲淡，降为低饱和度的淡色光。例如，一束高饱和度的蓝色光投射到屏幕上会被看成深蓝色光，若再将一束白色光也投射到屏幕上并与深蓝色重叠，则深蓝色变成淡蓝色，而且投射的白色光越强，颜色越淡，即饱和度越低。相反，由于在彩色电视的屏幕上的亮度过高，则饱和度降低，颜色被冲淡，这时可以降低亮度（白光）而使饱和度增大，颜色加深。

当彩色的饱和度降低时，其固有色彩特性也被降低和发生变化。例如，红色与绿色配置在一起，往往具有一种对比效果，但只有当红色与绿色都呈现饱和状态时，其对比效果才比较强烈。如果红色与绿色的饱和度都降低，红色变成浅红或暗红，绿色变成浅绿或深绿，再把它们配置在一起时相互的对比特征就会减弱，而趋于和谐。另外饱和度高的色彩容易让人感到单调刺眼。饱和度低，色感比较柔和协调，但混色太杂又容易让

人感觉浑浊，色调显得灰暗。

参考答案

（14）B

试题（15）、（16）

若用户需求不清晰且经常发生变化，但系统规模不太大且不太复杂，则最适宜采用＿（15）＿开发方法。对于数据处理领域的问题，若系统规模不太大且不太复杂，需求变化也不大，则最适宜采用＿（16）＿开发方法。

（15）A．结构化　　B．Jackson　　C．原型化　　D．面向对象

（16）A．结构化　　B．Jackson　　C．原型化　　D．面向对象

试题（15）、（16）分析

本题考查软件开发方法的基础知识。

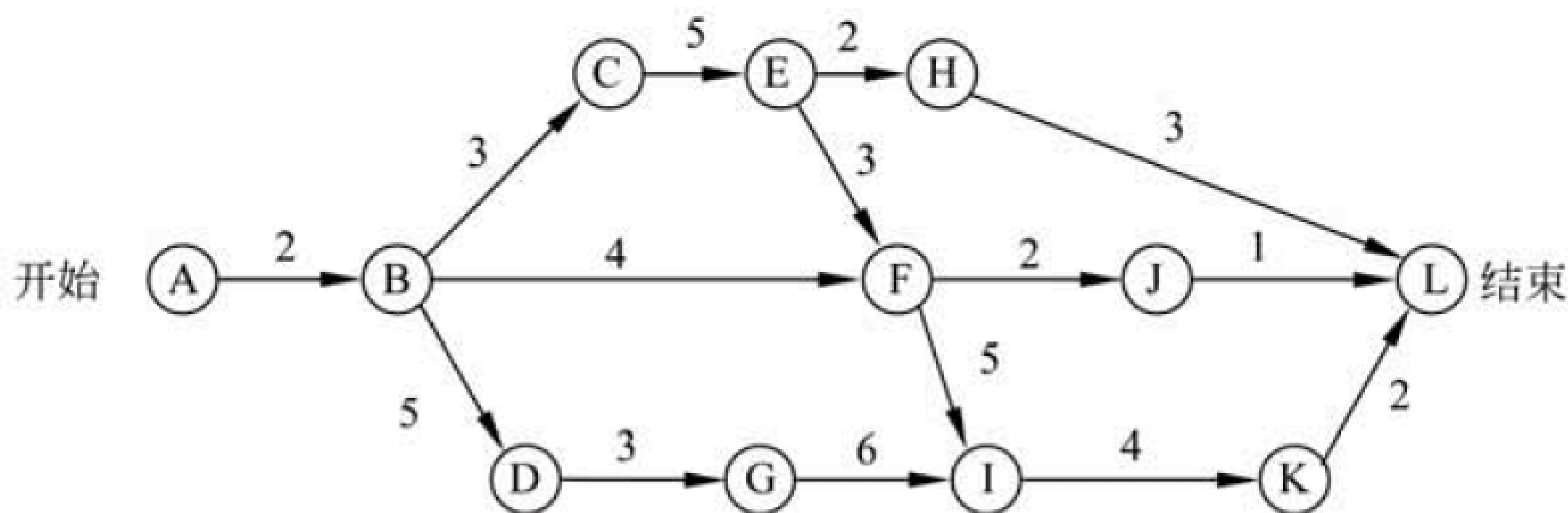
要求考生掌握典型的软件开发方法的基本概念和应用场合。需求不清晰且规模不太大时采用原型化方法最合适，而数据处理领域的不太复杂的软件，适于用结构化方法进行开发。

参考答案

（15）C　　（16）A

试题（17）、（18）

某软件项目的活动图如下图所示，其中顶点表示项目里程碑，连接顶点的边表示活动，边上的数字表示该活动所需的天数，则完成该项目的最少时间为＿（17）＿天。活动 BD 最多可以晚＿（18）＿天开始而不会影响整个项目的进度。



（17）A．9　　B．15　　C．22　　D．24

（18）A．2　　B．3　　C．5　　D．9

试题（17）、（18）分析

本题考查软件项目管理的基础知识。

根据上图计算出关键路径为 A-B-C-E-F-I-K-L，其长度为 24，关键路径上的活动均为关键活动。活动 BD 不在关键路径上，包含该活动的最长路径为 A-B-D-G-I-K-L，其长度为 22，因此松弛时间为 2。

参考答案

（17）D （18）A

试题（19）

以下关于软件项目管理中人员管理的叙述，正确的是__（19）__。

（19）A. 项目组成员的工作风格也应该作为组织团队时要考虑的一个要素
B. 鼓励团队的每个成员充分地参与开发过程的所有阶段
C. 仅根据开发人员的能力来组织开发团队
D. 若项目进度滞后于计划，则增加开发人员一定可以加快开发进度

试题（19）分析

本题考查软件项目管理的基础知识。

人员管理是软件项目管理的一个重要部分，在组织开发团队时，应该考虑开发人员的工作能力、知识背景、工作风格、兴趣爱好等多方面的因素。每个成员的工作任务分配清楚，不应该参与所有阶段的工作。当项目进度滞后于项目计划时，增加开发人员不一定可以加快开发进度。

参考答案

（19）A

试题（20）、（21）

编译器和解释器是两种基本的高级语言处理程序。编译器对高级语言源程序的处理过程可以划分为词法分析、语法分析、语义分析、中间代码生成、代码优化、目标代码生成等阶段，其中，__（20）__并不是每个编译器都必需的。与编译器相比，解释器__（21）__。

（20）A. 词法分析和语法分析 B. 语义分析和中间代码生成
C. 中间代码生成和代码优化 D. 代码优化和目标代码生成

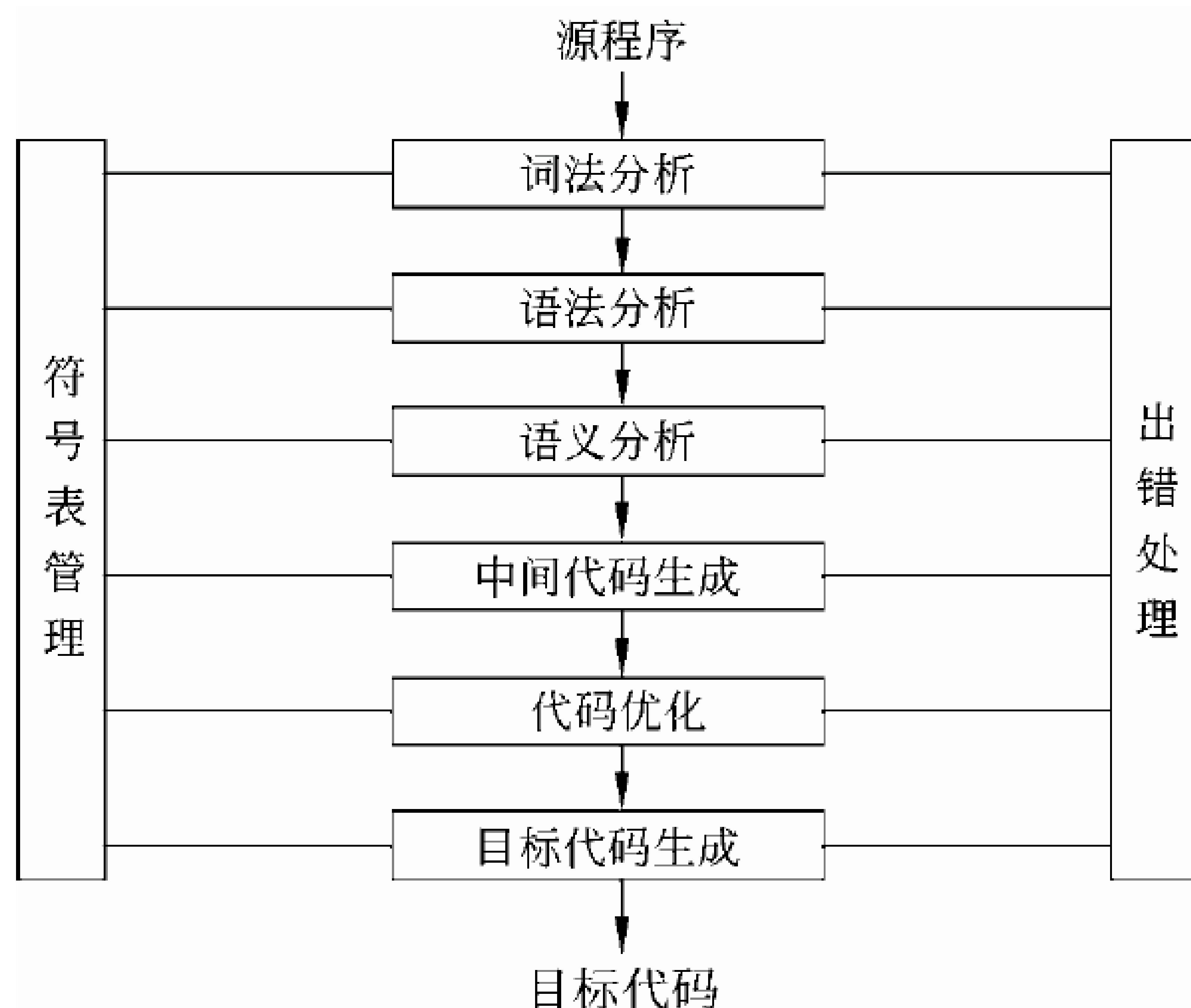
（21）A. 不参与运行控制，程序执行的速度慢
B. 参与运行控制，程序执行的速度慢
C. 参与运行控制，程序执行的速度快
D. 不参与运行控制，程序执行的速度快

试题（20）、（21）分析

本题考查程序语言基础知识。

解释程序也称为解释器，它可以直接解释执行源程序，或者将源程序翻译成某种中间表示形式后再加以执行；而编译程序（编译器）则首先将源程序翻译成目标语言程序，然后在计算机上运行目标程序。这两种语言处理程序的根本区别是：在编译方式下，机器上运行的是与源程序等价的目标程序，源程序和编译程序都不再参与目标程序的执行过程；而在解释方式下，解释程序和源程序（或其某种等价表示）要参与到程序的运行过程中，运行程序的控制权在解释程序。解释器翻译源程序时不产生独立的目标程序，而编译器则需将源程序翻译成独立的目标程序。

分阶段编译器的工作过程如下图所示。其中，中间代码生成和代码优化不是必须的。



参考答案

（20）C　　　（21）B

试题（22）

表达式采用逆波兰式表示时，利用__（22）__进行求值。

（22）A．栈　　　B．队列　　　C．符号表　　　D．散列表

试题（22）分析

本题考查程序语言基础知识。

后缀式（逆波兰式）是波兰逻辑学家卢卡西维奇发明的一种表示表达式的方法。这种表示方式把运算符写在运算对象的后面，例如，把 a+b 写成 ab+，所以也称为后缀式。

借助栈可以方便地对后缀式进行求值。方法为：先创建一个初始为空的栈，用来存放运算数。对后缀表达式求值时，从左至右扫描表达式，若遇到运算数，就将其入栈，若遇到运算符，就从栈顶弹出需要的运算数并进行运算，然后将结果压入栈顶，如此重复，直到表达式结束。若表达式无错误，则最后的运算结果就存放在栈顶并且是栈中唯一的元素。

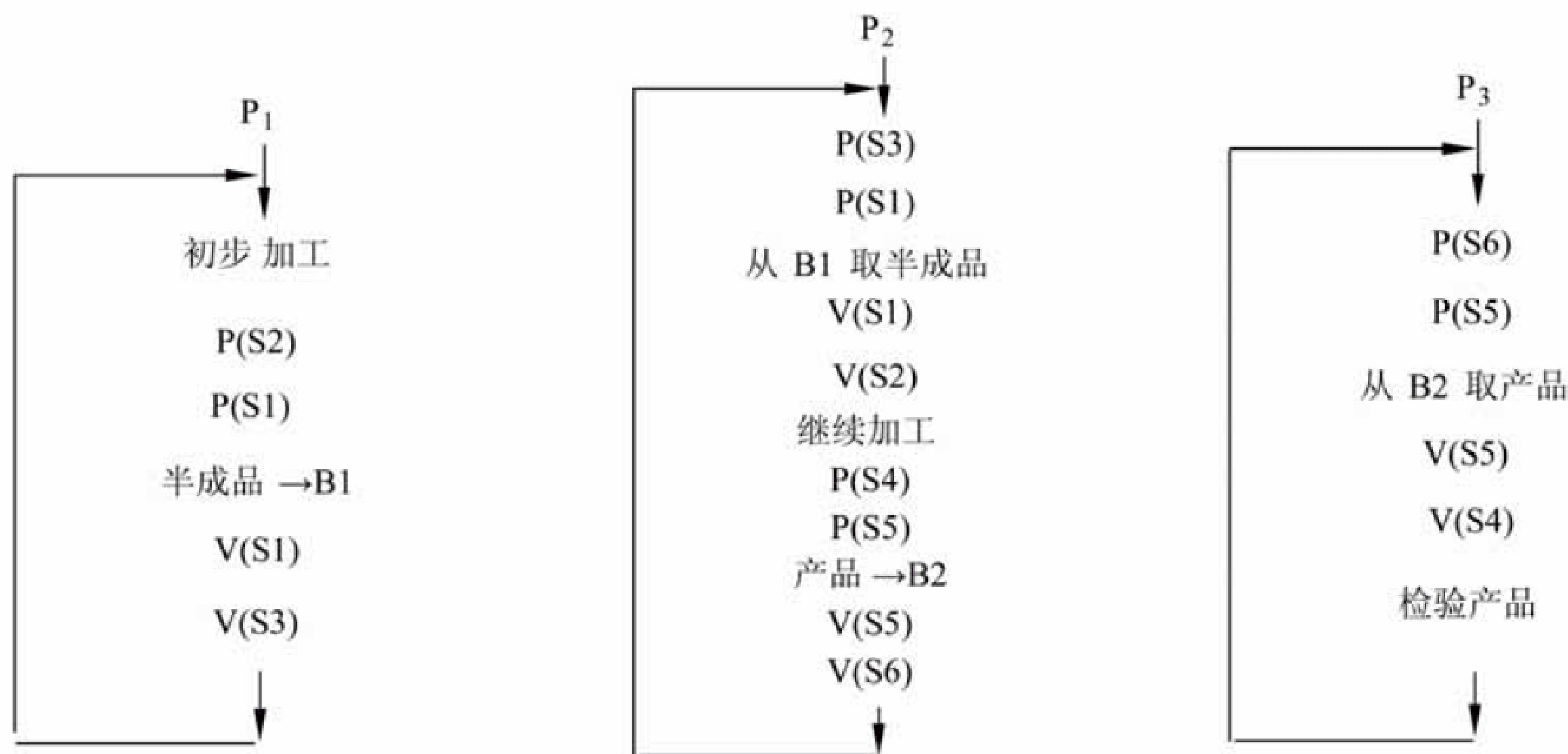
参考答案

（22）A

试题（23）、（24）

某企业的生产流水线上有 2 名工人 P_1 和 P_2，1 名检验员 P_3。P_1 将初步加工的半成品放入半成品箱 B1；P_2 从半成品箱 B1 取出继续加工，加工好的产品放入成品箱 B2；P_3 从成品箱 B2 取出产品检验。假设 B1 可存放 n 件半成品，B2 可存放 m 件产品，并设置 6 个信号量 S1、S2、S3、S4、S5 和 S6，且 S3 和 S6 的初值都为 0。采用 PV 操作实现 P_1、

P_2 和 P_3 的同步模型如下图所示，则信号量 S1 和 S5 （23） ；S2、S4 的初值分别为 （24） 。



（23）A. 分别为同步信号量和互斥信号量，初值分别为 0 和 1
B. 都是同步信号量，其初值分别为 0 和 0
C. 都是互斥信号量，其初值分别为 1 和 1
D. 都是互斥信号量，其初值分别为 0 和 1

（24）A. n、0　　B. m、0　　C. m、n　　D. n、m

试题（23）、（24）分析

试题（23）的正确的答案为 C。因为信号量 S1 是一个互斥信号量，表示半成品箱 B1 当前有无工人（生产者）使用，所以初值为 1。信号量 S5 也是一个互斥信号量，表示成品箱 B2 当前有无工人或检验员使用，所以初值为 1。

试题（24）的正确的答案为 D。信号量 S2 表示半成品箱 B1 的容量，故 S2 的初值为 *n*。当工人 P1 不断地将其工序上加工的半成品放入半成品箱 B1 时，应该先测试半成品箱是否有空位，故工人 P1 使用 P(S2)，当工人 P2 从半成品箱取一件半成品时，半成品箱 B1 就空出一个空位，故工人 P2 使用 V(S2)释放空间。

同理，信号量 S4 表示成品箱 B2 的容量，故 S4 的初值为 m。当工人 P2 完成一件产品放入成品箱 B2 时，应该先测试成品箱是否有空位，故工人 P2 使用 P(S4)，当检验员 P3 从成品箱取一件产品检验时，成品箱 B2 就空出一个空位，故检验员 P3 使用 V(S4)释放空间。

参考答案

（23）C　（24）D

试题（25）、（26）

假设磁盘块与缓冲区大小相同，每个盘块读入缓冲区的时间为 15μs，由缓冲区送至用户区的时间是 5μs，在用户区内系统对每块数据的处理时间为 1μs。若用户需要将大小

为 10 个磁盘块的 Doc1 文件逐块从磁盘读入缓冲区，并送至用户区进行处理，那么采用单缓冲区需要花费的时间为__(25)__μs；采用双缓冲区需要花费的时间为__(26)__μs。

(25) A. 150　　B. 151　　C. 156　　D. 201

(26) A. 150　　B. 151　　C. 156　　D. 201

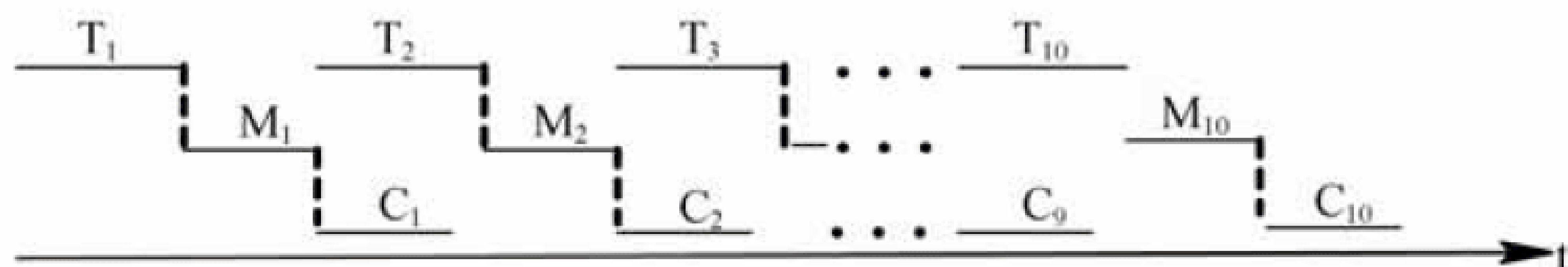
试题（25）、（26）分析

试题（25）的正确的答案为 D。在块设备输入时，假定从磁盘把一块数据输入到缓冲区的时间为 T，缓冲区中的数据传送到用户工作区的时间为 M，而系统处理（计算）的时间为 C，如图（a）所示。



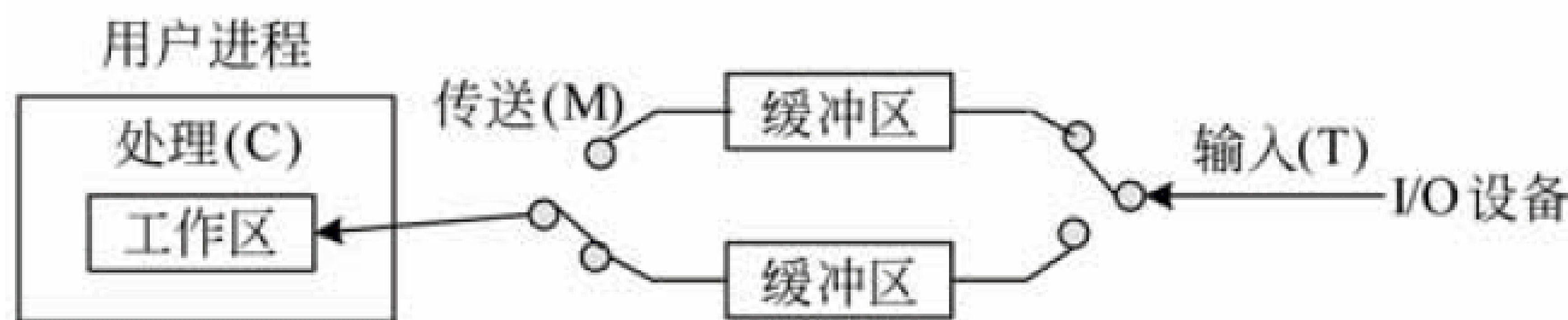
图（a）单缓冲工作过程图

当第一块数据送入用户工作区后，缓冲区是空闲的就可以传送第二块数据。这样第一块数据的处理 C_1 与第二块数据的输入 T_2 是可以并行的，依次类推，如图（b）所示。系统对每一块数据的处理时间为：Max(C，T)+M。因为，当 T>C 时，处理时间为 M+T；当 T<C 时，处理时间为 M+C。本题每一块数据的处理时间为 15+5=20，Doc1 文件的处理时间为 20*10+1。



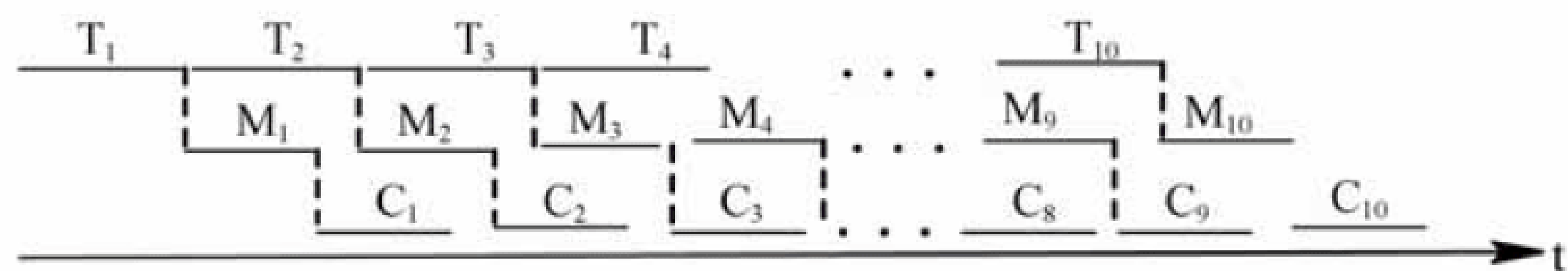
图（b）单缓冲并行工作示意图

试题（26）的正确的答案为 C。双缓冲工作方式基本方法是在设备输入时，先将数据输入到缓冲区 1，装满后便转向缓冲区 2。此时系统可以从缓冲区 1 中提取数据传送到用户区，最后由系统对数据进行处理，如图（c）所示。



图（c）双缓冲工作过程图

双缓冲可以实现对缓冲区中数据的输入 T 和提取 M，与 CPU 的计算 C，三者并行工作，如图（d）所示。从图中可以看出，双缓冲进一步加快了 I/O 的速度，提高了设备的利用率。在双缓冲时，系统处理一块数据的时间可以粗略地认为是 Max(C，T)。如果 C<T，可使块设备连续输入；如果 C>T，则可使系统不必等待设备输入。本题每一块数据的处理时间为 10，采用双缓冲需要花费的时间为 15*10+5+1=156。



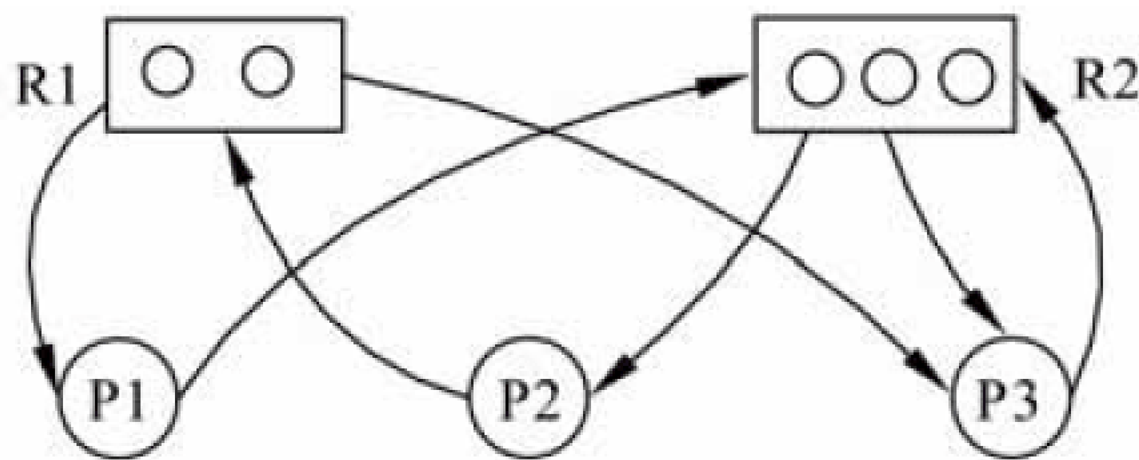
图(d) 双缓冲并行工作示意图

参考答案

（25）D （26）C

试题（27）

在如下所示的进程资源图中， （27） 。



（27）A．P1、P2、P3 都是非阻塞节点，该图可以化简，所以是非死锁的

B．P1、P2、P3 都是阻塞节点，该图不可以化简，所以是死锁的

C．P1、P2 是非阻塞节点，P3 是阻塞节点，该图不可以化简，所以是死锁的

D．P2 是阻塞节点，P1、P3 是非阻塞节点，该图可以化简，所以是非死锁的

试题（27）分析

R2 资源有 3 个，已分配 2 个，P3 申请 1 个 R2 资源可以得到满足，故进程 P3 可以运行完毕释放其占有的资源。这样可以使得 P1、P2 都变为非阻塞节点，得到所需资源运行完毕，因此，该进程资源图是可化简的。

参考答案

（27）D

试题（28）

在支持多线程的操作系统中，假设进程 P 创建了若干个线程，那么 （28） 是不能被这些线程共享的。

（28）A．该进程中打开的文件 B．该进程的代码段

C．该进程中某线程的栈指针 D．该进程的全局变量

试题（28）分析

在同一进程中的各个线程都可以共享该进程所拥有的资源，如访问进程地址空间中的每一个虚地址；访问进程所拥有的已打开文件、定时器、信号量机构等，但是不能共享进程中某线程的栈指针。

参考答案

（28）C

试题（29）

某开发小组欲开发一个超大规模软件：使用通信卫星，在订阅者中提供、监视和控制移动电话通信，则最不适宜采用__（29）__过程模型。

（29）A．瀑布　B．原型　C．螺旋　D．喷泉

试题（29）分析

本题考查软件开发过程模型的基础知识。

瀑布模型将开发阶段描述为从一个阶段瀑布般地转换到另一个阶段的过程。

原型模型中，开发人员快速地构造整个系统或者系统的一部分以理解或澄清问题。

螺旋模型将开发活动和风险管理结合起来，以减小风险。

喷泉模型开发过程模型以用户需求为动力，以对象为驱动，适合于面向对象的开发方法。

在这几种开发过程模型中，原型模型不适宜大规模软件的开发。

参考答案

（29）B

试题（30）

__（30）__开发过程模型以用户需求为动力，以对象为驱动，适合于面向对象的开发方法。

（30）A．瀑布　B．原型　C．螺旋　D．喷泉

试题（30）分析

本题考查软件开发过程模型的基础知识。

根据题干描述，合适的开发过程模型为喷泉模型。

参考答案

（30）D

试题（31）

在 ISO/IEC 软件质量模型中，易使用性的子特性不包括__（31）__。

（31）A．易理解性　B．易学性　C．易操作性　D．易分析性

试题（31）分析

本题考查软件质量的基础知识。

ISO/IEC 软件质量模型由三个层次组成：第一层是质量特性，第二层是质量子特性，第三层是度量指标。易使用性是指与为使用所需的努力和由一组规定或隐含的用户对这样使用所做的个别评价有关的一组属性。其子特性包括易理解性、易学性、易操作性。

参考答案

（31）D

试题（32）

在进行子系统结构设计时，需要确定划分后的子系统模块结构，并画出模块结构图。

该过程不需要考虑__(32)__。

(32) A. 每个子系统如何划分成多个模块
B. 每个子系统采用何种数据结构和核心算法
C. 如何确定子系统之间、模块之间传送的数据及其调用关系
D. 如何评价并改进模块结构的质量

试题(32)分析

本题考查软件设计的基础知识。

子系统结构设计中,重点关注如何划分模块,子系统之间以及模块之间的数据和调用关系,模块结构质量等这些粗粒度的问题;而对每个模块内部进行设计时,才需要考虑采用的数据结构以及处理的算法。

参考答案

(32) B

试题(33)

数据流图中某个加工的一组动作依赖于多个逻辑条件的取值,则用__(33)__能够清楚地表示复杂的条件组合与应做的动作之间的对应关系。

(33) A. 流程图 B. NS 盒图 C. 形式语言 D. 决策树

试题(33)分析

本题考查结构化分析方面的基础知识。

在结构化分析中,用数据流图对软件功能建模,加工是数据流的一个重要要素,可以用多种方式描述,如流程图、NS 盒图等,其中决策树和决策表适于用来表示加工中涉及多个逻辑条件的情况。

参考答案

(33) D

试题(34)

根据软件过程活动对软件工具进行分类,则逆向工程工具属于__(34)__工具。

(34) A. 软件开发 B. 软件维护 C. 软件管理 D. 软件支持

试题(34)分析

本题考查软件工程过程及软件工具的基础知识。

逆向工程从源代码得到软件系统的规格说明和设计信息,属于软件维护阶段行为,因此逆向工程工具属于软件维护工具。

参考答案

(34) B

试题(35)、(36)

若用白盒测试方法测试以下代码,并满足条件覆盖,则至少需要__(35)__个测试用例。采用 McCabe 度量法算出该程序的环路复杂性为__(36)__。

```
int find_max(int i, int j, int k){
        int max;
        if (i > j) then
            if (i > k) then max = i;
            else max = k;
        else if (j > k) max = j;
            else max = k;
        return max;
}
```

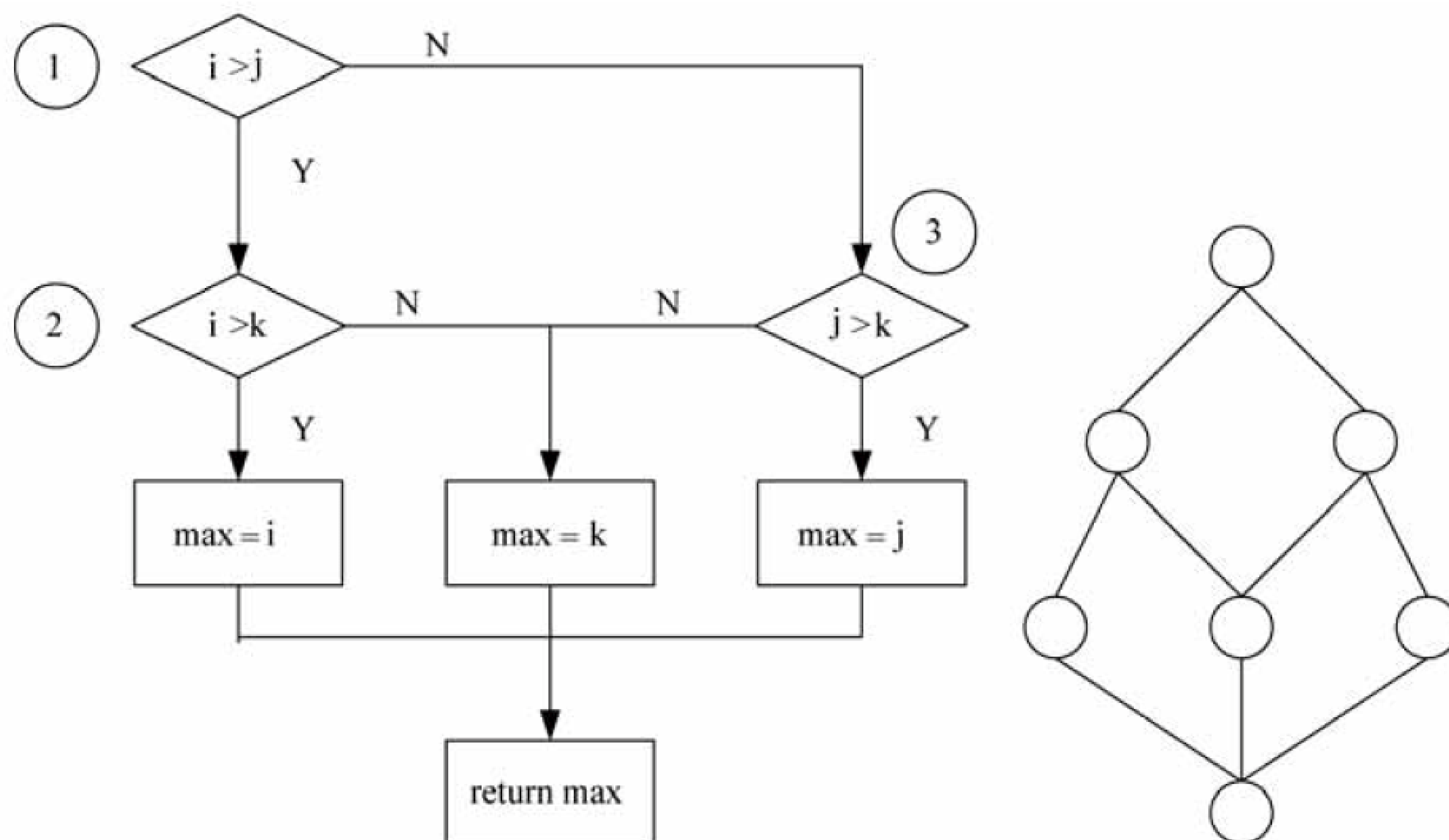
（35）A. 3　　B. 4　　C. 5　　D. 6

（36）A. 1　　B. 2　　C. 3　　D. 4

试题（35）、（36）分析

本题考查软件测试的基础知识。

画出该代码的流程图，如下：



要满足条件覆盖，要求三个判断框的 Y 和 N 至少要经过一次，即 1Y2Y；1Y2N；1N3Y；1N3N，至少需要 4 个测试用例。

McCabe 度量法是一种基于程序控制流的复杂性度量方法，环路复杂性为 $V(G) = m - n + 2$，图中 m=9，n=7，$V(G) = 9-7+2=4$。

参考答案

（35）B　　（36）D

试题（37）、（38）

在面向对象的系统中，对象是运行时实体，其组成部分不包括＿（37）＿；一个类定义了一组大体相似的对象，这些对象共享＿（38）＿。

（37）A．消息　　B．行为（操作）　　C．对象名　　D．状态

（38）A．属性和状态　　B．对象名和状态

　　C．行为和多重度　　D．属性和行为

试题（37）、（38）分析

本题考查面向对象的基本知识。

在面向对象系统中，对象是基本的运行时的实体，它既包括数据（属性），也包括作用于数据的操作（行为）。所以，一个对象把属性和行为封装为一个整体。封装是一种信息隐蔽技术，它的目的是使对象的使用者和生产者分离，使对象的定义和实现分开。从程序设计者来看，对象是一个程序模块；从用户来看，对象为他们提供了所希望的行为。在对象内的操作通常叫做方法。一个对象通常可由对象名、属性和方法三部分组成。

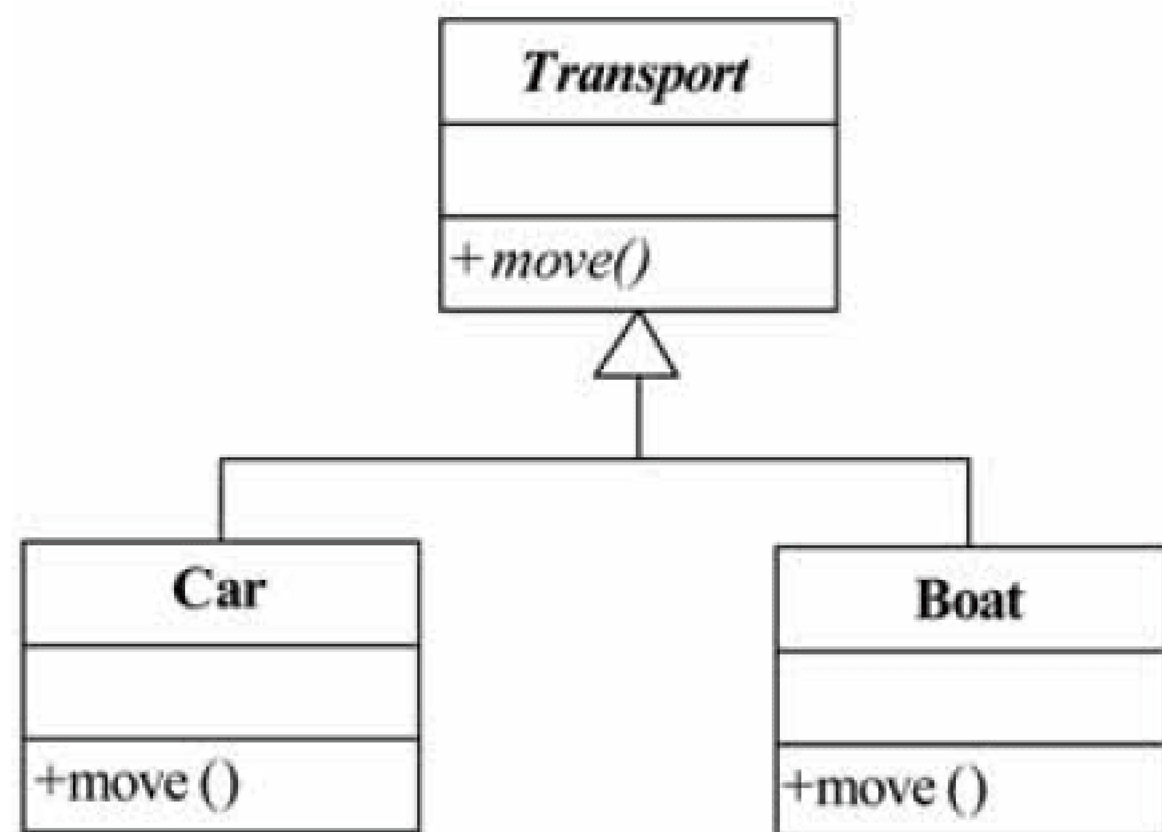
一个类定义了一组大体上相似的对象。一个类所包含的方法和数据描述一组对象的共同行为和属性，这些对象共享这些行为和属性。

参考答案

（37）A　　（38）D

试题（39）

如下所示的 UML 类图中，Car 和 Boat 类中的 move()方法__（39）__了 Transport 类中的 move()方法。



（39）A．继承　　B．覆盖（重置）　　C．重载　　D．聚合

试题（39）分析

本题考查面向对象和统一建模语言（UML）的基本知识。

一个类定义了一组大体上相似的对象。一个类所包含的方法和数据描述一组对象的共同行为和属性。有些类之间存在一般和特殊关系，即一些类是某个类的特殊情况，某个类是一些类的一般情况，即继承关系。继承是父类和子类之间共享数据和方法的机制。父类描述了这些子类的公共属性和方法。一个子类可以继承它的父类（或祖先类）中的属性和方法，这些属性和操作在子类中不必定义，子类中还可以定义自己的属性和方法，也可以重新定义父类中已经定义的方法，即重置或覆盖（overriding）。UML 类图中，如果父类中已有方法名在子类中不出现，表示子类继承父类中的方法；如果父类中已有方法名在子类中出现了，就表示子类在继承父类接口定义的前提下，用适合于自己要求的

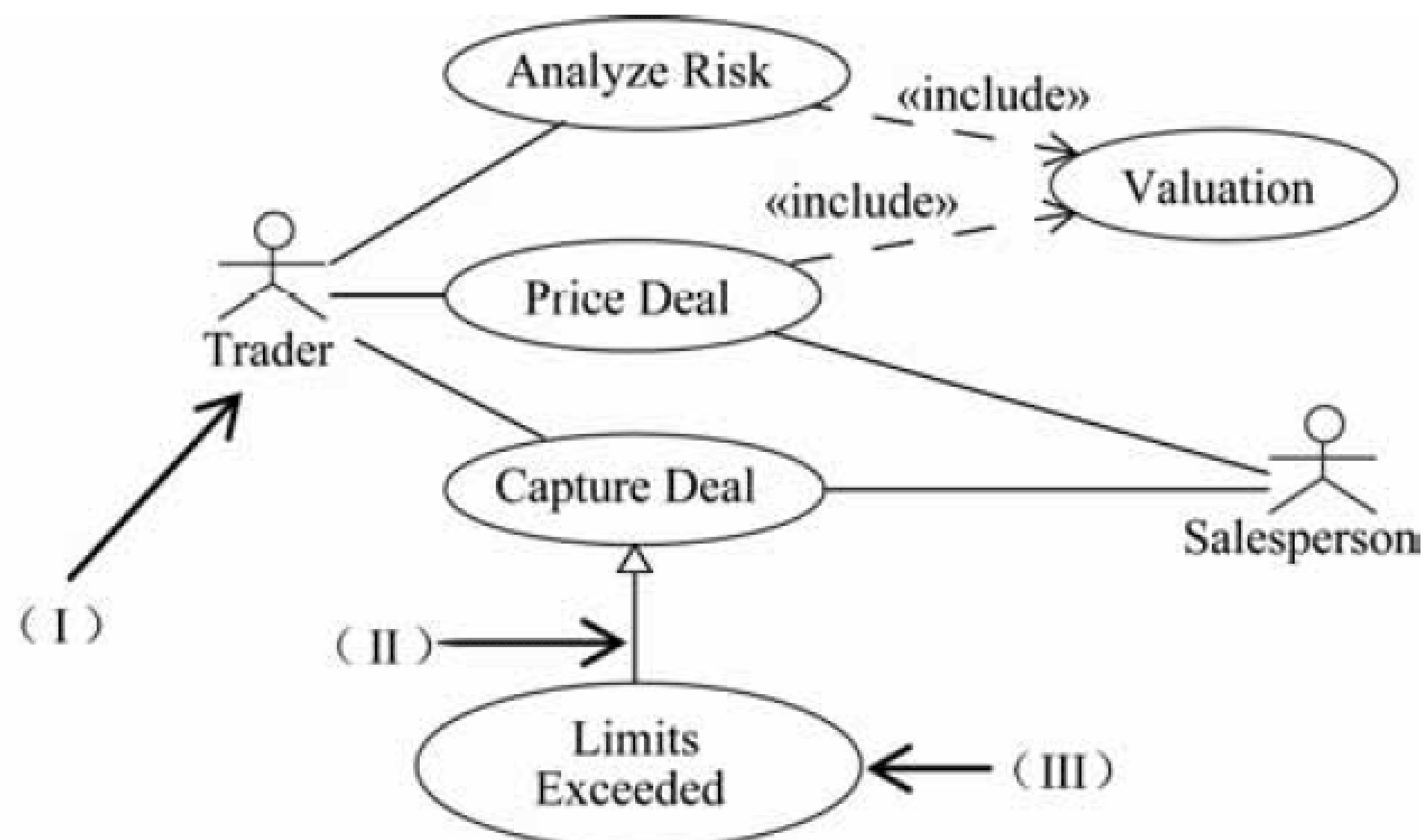
实现去置换父类中的相应实现，即覆盖了父类中的方法。

参考答案

（39）B

试题（40）～（42）

如下所示的 UML 图中，（I）是__（40）__，（II）是__（41）__，（III）是__（42）__。



（40）A. 参与者　B. 用例　C. 泛化关系　D. 包含关系

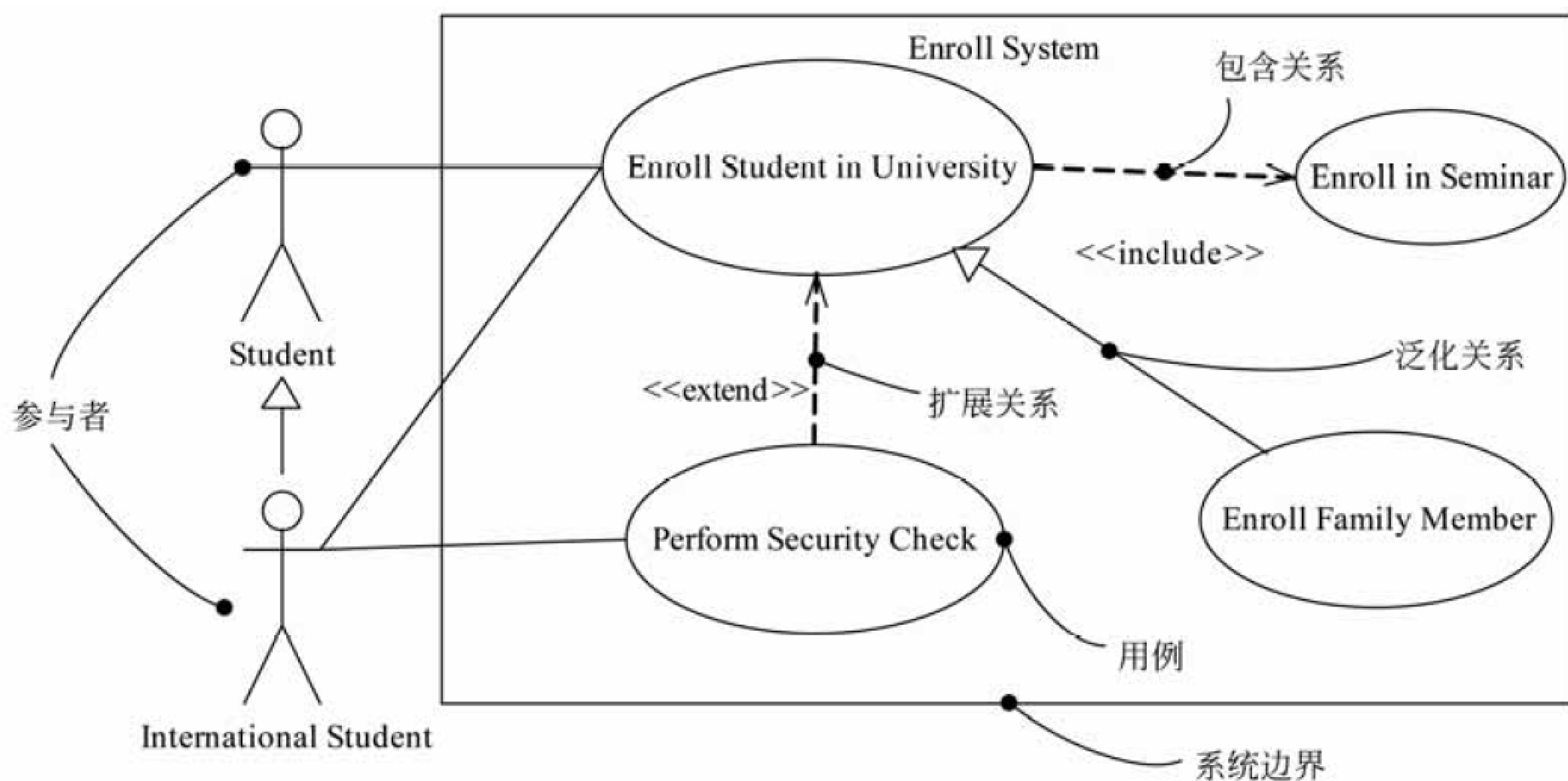
（41）A. 参与者　B. 用例　C. 泛化关系　D. 包含关系

（42）A. 参与者　B. 用例　C. 泛化关系　D. 包含关系

试题（40）～（42）分析

本题考查统一建模语言（UML）的基本知识。

用例图（use case diagram）展现了一组用例、参与者（Actor）以及它们之间的关系。用例图通常包括用例、参与者，以及用例之间的扩展关系（<<extend>>）和包含关系（<<include>>），参与者和用例之间的关联关系，用例与用例以及参与者与参与者之间的泛化关系。如下图所示。



用例图用于对系统的静态用例视图进行建模，主要支持系统的行为，即该系统在它的周边环境的语境中所提供的外部可见服务。

参考答案

（40）A （41）C （42）B

试题（43）

下图所示为 UML__（43）__。



（43）A．类图 B．部署图 C．组件图 D．网络图

试题（43）分析

本题考查统一建模语言（UML）的基本知识。

UML 中提供了多种建模系统的图，体现系统的静态方面和动态方面。类图（class diagram）展现了一组对象、接口、协作和它们之间的关系。在面向对象系统的建模中所建立的最常见的图就是类图。类图给出系统的静态设计视图。部署图（deployment diagram）是用来对面向对象系统的物理方面建模的方法，展现了运行时处理结点以及其中构件（制品）的配置。部署图对系统的静态部署视图进行建模，它与组件图（构件图）相关。组件图或构件图（component diagram）展现了一组构件之间的组织和依赖，如题中的图所示。组件图或构件图专注于系统的静态实现视图。它与类图相关，通常把构件映射为一个或多个类、接口或协作。UML 部署图经常被认为是一个网络图。

参考答案

（43）C

试题（44）

以下关于 Singleton（单例）设计模式的叙述中，不正确的是__（44）__。

（44）A．单例模式是创建型模式

B．单例模式保证一个类仅有一个实例

C．单例类提供一个访问唯一实例的全局访问点

D．单例类提供一个创建一系列相关或相互依赖对象的接口

试题（44）分析

本题考查设计模式的基本概念。

Singleton（单例）设计模式是一种创建型模式，其意图是保证一个类仅有一个实例，并提供一个访问这个唯一实例的全局访问点。单例模式适用于当类只能有一个实例而且客户可以从一个众所周知的访问点访问它时，以及当这个唯一实例应该是通过子类化可扩展的，并且客户应该无须更改代码就能使用一个扩展的实例时。

参考答案

（44）D

试题（45）～（47）

__（45）__设计模式能够动态地给一个对象添加一些额外的职责而无须修改此对象的结构；__（46）__设计模式定义一个用于创建对象的接口，让子类决定实例化哪一个类；欲使一个后端数据模型能够被多个前端用户界面连接，采用__（47）__模式最适合。

（45）A. 组合（Composite）　B. 外观（Façade）
C. 享元（Flyweight）　D. 装饰器（Decorator）

（46）A. 工厂方法（Factory Method）　B. 享元（Flyweight）
C. 观察者（Observer）　D. 中介者（Mediator）

（47）A. 装饰器（Decorator）　B. 享元（Flyweight）
C. 观察者（Observer）　D. 中介者（Mediator）

试题（45）～（47）分析

本题考查设计模式的基本概念。每种设计模式都有特定的意图，描述一个在我们周围不断重复发生的问题，以及该问题的解决方案的核心，使该方案能够重用而不必做重复劳动。

组合（Composite）模式将对象组合成树形结构以表示“部分-整体”的层次结构，使得用户对单个对象和组合对象的使用具有一致性。适用于：想表示对象的部分-整体层次结构；希望用户忽略组合对象与单个对象的不同，用户将统一地使用组合结构中的所有对象。

外观（Facade）模式为子系统中的一组接口提供一个一致的界面，Façade 模式定义了一个高层接口，这个接口使得这一子系统更加容易使用。适用于：要为一个复杂子系统提供一个简单接口时，子系统往往因为不断演化而变得越来越复杂；客户程序与抽象类的实现部分之间存在着很大的依赖性；当需要构建一个层次结构的子系统时，使用 Façade 模式定义子系统中每层的入口点。

享元（Flyweight）模式运用共享技术有效地支持大量细粒度的对象。适用于：一个应用程序使用了大量的对象；完全由于使用大量的对象，造成很大的存储开销；对象的大多数状态都可变为外部状态；如果删除对象的外部状态，那么可以用相对较少的共享对象取代很多组对象；应用程序不依赖于对象标识。

装饰器（Decorator）模式描述了以透明围栏来支持修饰的类和对象的关系，动态地给一个对象添加一些额外的职责，从增加功能的角度来看，装饰器模式相比生成子类更

加灵活。适用于：在不影响其他对象的情况下，以动态、透明的方式给单个对象添加职责；处理那些可以撤销的职责；当不能采用生成子类的方式进行扩充时。

工厂方法（Factory Method）定义一个用于创建对象的接口，让子类决定将哪一个类实例化，使一个类的实例化延迟到其子类。适用于：当一个类不知道它所必须创建的对象的类的时候；当一个类希望由它的子类来指定它所创建的对象的时候；当类将创建对象的职责委托给多个帮助子类中的某一个，并且希望将哪一个帮助子类是代理者这一信息局部化的时候。

观察者（Observer）模式定义对象间的一种一对多的依赖关系，当一个对象的状态发生改变时，所有依赖于它的对象都得到通知并被自动更新。适用于：当一个抽象模型有两个方面，其中一个方面依赖于另一个方面，将这两者封装在独立的对象中以使它们可以各自独立地改变和复用；当对一个对象的改变需要同时改变其他对象，而不知道具体有多少对象有待改变时；当一个对象必须通知其他对象，而它又不能假定其他对象是谁，即不希望这些对象是紧耦合的。

中介者（Mediator）用一个中介对象来封装一系列的对象交互。中介者使各对象不需要显式地相互引用，从而使其耦合松散，而且可以独立地改变它们之间的交互。适用于：一组对象以定义良好但是复杂的方式进行通信，产生的相互依赖关系结构混乱且难以理解；一个对象引用其他很多对象并且直接与这些对象通信，导致难以复用该对象；想定制一个分布在多个类中的行为，而又不想生成太多的子类。如使一个后端数据模型能够被多个前端用户界面连接，采用此模式最合适。

参考答案

（45）D　　（46）A　　（47）D

试题（48）

某程序运行时陷入死循环，则可能的原因是程序中存在__（48）__。

（48）A. 词法错误　　B. 语法错误

C. 动态的语义错误　　D. 静态的语义错误

试题（48）分析

本题考查程序语言基础知识。

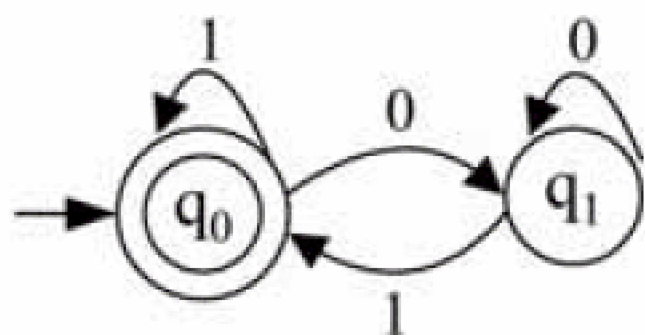
程序已经开始运行，说明编译时无错误，因此不是语法错误和词法错误，编译时发现的语义错误称为静态的语义错误。运行时陷入死循环属于动态语义错误。

参考答案

（48）C

试题（49）

某非确定的有限自动机（NFA）的状态转换图如下图所示（q_0 既是初态也是终态）。以下关于该 NFA 的叙述中，正确的是__（49）__。



（49）A. 其可识别的 0、1 序列的长度为偶数

B. 其可识别的 0、1 序列中 0 与 1 的个数相同

C. 其可识别的非空 0、1 序列中开头和结尾字符都是 0

D. 其可识别的非空 0、1 序列中结尾字符是 1

试题（49）分析

本题考查程序语言基础知识。

若存在一条从初态到某一终止状态的路径，且这条路径上所有弧的标记符连接成的字符串等于 ω，则称 ω 可由 NFA 识别（接受或读出）。

对于题中给出的 NFA，其初态为 q_0，q_0 上的自回路表示识别零个或多个 1，接下来识别出一个 0 时进入状态 q_1，q_1 上的自回路表示识别零个或多个 0，接下来识别出 1 个 1 之后再回到 q_0。

例如，该自动机可识别空串（因为 q_0 既是初态，也是终态）、01、00001、101、1、11、111、1111 等。

01 的识别路径为 q_0->q_1->q_0

00001 的识别路径为 q_0->q_1->q_1->q_1->q_1->q_0

101 的识别路径为 q_0->q_0->q_1->q_0

1 的识别路径为 q_0->q_0

11 的识别路径为 q_0->q_0->q_0

111 的识别路径为 q_0->q_0->q_0->q_0

1111 的识别路径为 q_0->q_0->q_0->q_0->q_0

识别字符串时必须从初始状态 q_0 出发，并回到状态 q_0，因此对于仅由 1 构成的任意长度的串，在识别过程中不会离开 q_0。当识别出一个 0 而离开 q_0 后就进入 q_1，此后的字符若全部为 0，则会一直在 q_1，直到识别出一个 1 而回到 q_0，因此除了空串，该 NFA 识别的字符串必须以 1 结尾。

参考答案

（49）D

试题（50）

函数 t()、f()的定义如下所示，若调用函数 t 时传递给 x 的值为 5，并且调用函数 f()时，第一个参数采用传值（call by value）方式，第二个参数采用传引用（call by reference）方式，则函数 t 的返回值为<u>（50）</u>。

```
t(int x)
int a;
a = 3*x + 1;
f(x, a);
return a-x;
```

```
f(int r, int &s)
int x;
x = 2*s +1 ; s = x+r;
r = x - 1;
return;
```

（50）A．33　　B．22　　C．11　　D．负数

试题（50）分析

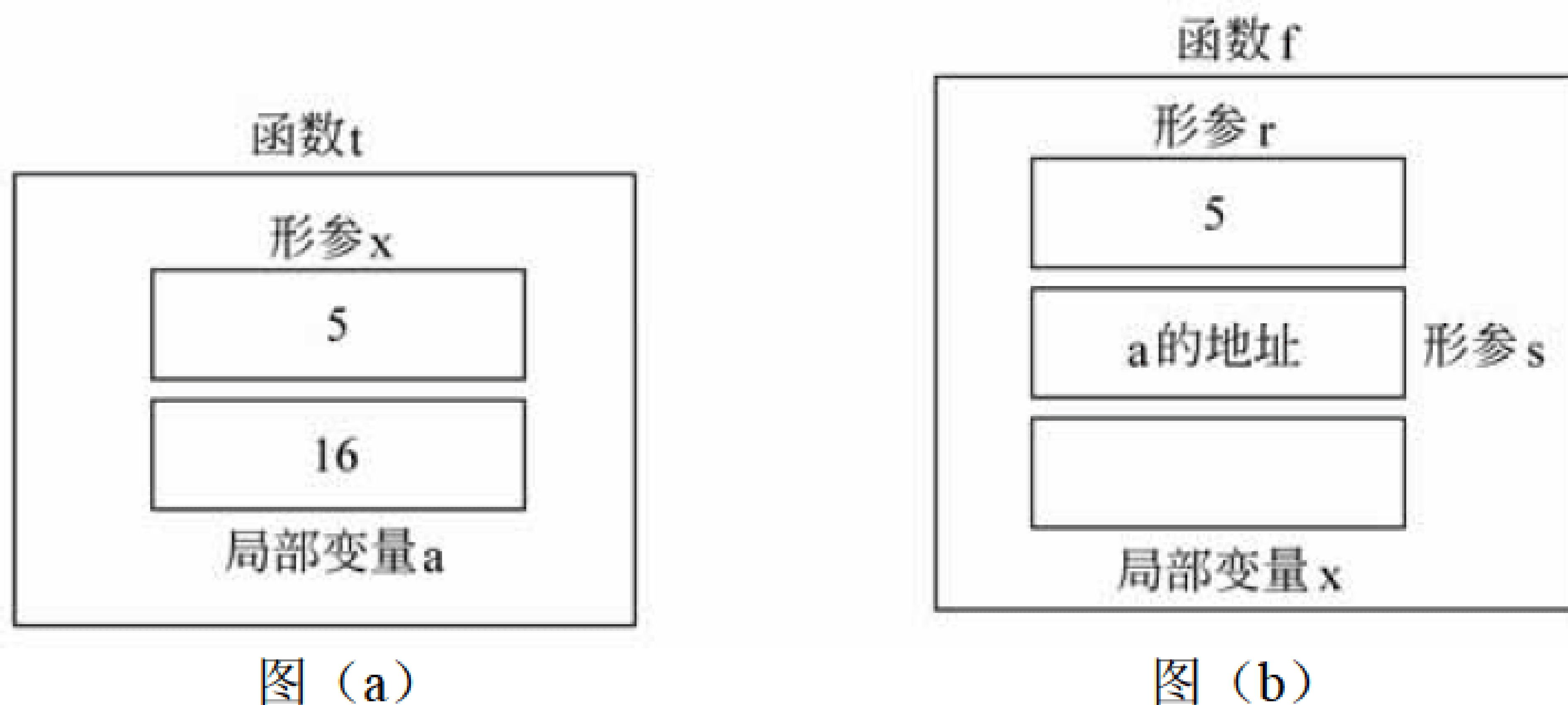
本题考查程序语言基础知识。

若函数调用时采用传值方式，则是将实参的值传给形参，再执行被调用的函数，对形参的修改不影响实参。若采用传引用方式，则是将实参的地址传递给形参，本质上是通过间接访问的方式修改实参，也可以简化理解为：在被调用函数中对形参的修改等同于是对实参进行修改。

当函数 t 和 f 运行时，其每个形参和局部变量都有各自的存储单元，下面图中矩形框表示存储单元。

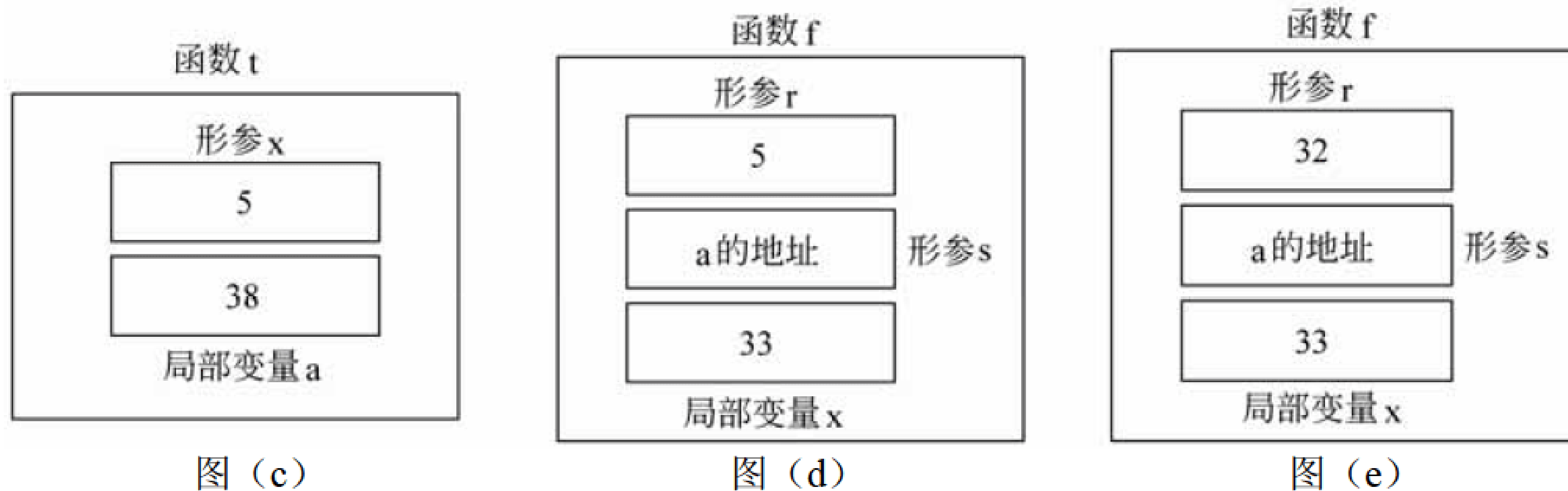
如题中所述，调用 t 时传递给其形参 x 的值为 5。因此函数 t 被调用而执行时，在执行函数调用 f(x, a)之前，其形参 x 和局部变量 a 的值如下图（a）所示。

执行函数调用 f(x, a)时，t 中 x 的值传给 f 的形参 r、a 的地址传给 f 的形参 s，如下图（b）所示。



图（a）　　图（b）

在 f 执行时，其局部变量 x 的值由运算“x = 2*s +1”改为 33，其中运算时可理解为 s 提供了 t 的局部变量 a 的值（是由间接访问机制实现的）。接下来的运算为“s = x+r”，也就是将 x 的值（即 33）与 r 的值（即 5）相加得到 38，然后（通过间接访问）存入 t 的局部变量 a，结果如下图（c）、（d）所示。之后运算“r = x - 1”将 r 的值改为 32，结果如下图（e）所示。



图（c）　　图（d）　　图（e）

当函数 f 运行结束并返回函数 t 后，函数 f 的运行空间将由系统撤销，接下来运算“a-x”产生的值为 33（即 38-5），因此函数 t 的返回值为 33。

参考答案

（50）A

试题（51）

数据库系统通常采用三级模式结构：外模式、模式和内模式。这三级模式分别对应数据库的__（51）__。

（51）A．基本表、存储文件和视图　　B．视图、基本表和存储文件
C．基本表、视图和存储文件　　D．视图、存储文件和基本表

试题（51）分析

本题考查数据库的基本概念。

数据库通常采用三级模式结构，其中，视图对应外模式、基本表对应模式、存储文件对应内模式。

参考答案

（51）B

试题（52）

在数据库逻辑设计阶段，若实体中存在多值属性，那么将 E-R 图转换为关系模式时，__（52）__，得到的关系模式属于 4 NF。

（52）A．将所有多值属性组成一个关系模式
B．使多值属性不在关系模式中出现
C．将实体的码分别和每个多值属性独立构成一个关系模式
D．将多值属性和其他属性一起构成该实体对应的关系模式

试题（52）分析

本题考查对数据库应用系统设计中逻辑结构设计的掌握。

在数据库设计中，将 E-R 图转换为关系模式是逻辑设计的主要内容。转换中将实体转换为关系模式，对实体中的派生属性不予考虑，组合属性只取各组合分量，若不含多值属性，通常一个实体对应一个关系模式。对实体中的多值属性，取实体的码和多值属性构成新增的关系模式，且该新增关系模式中，实体的码多值决定多值属性，属于平凡的多值依赖，关系属于 4NF。

参考答案

（52）C

试题（53）、（54）

在分布式数据库中有分片透明、复制透明、位置透明和逻辑透明等基本概念，其中：__（53）__是指局部数据模型透明，即用户或应用程序无须知道局部使用的是哪种数据模型；__（54）__是指用户或应用程序不需要知道逻辑上访问的表具体是如何分块存储的。

（53）A. 分片透明　B. 复制透明　C. 位置透明　D. 逻辑透明

（54）A. 分片透明　B. 复制透明　C. 位置透明　D. 逻辑透明

试题（53）、（54）分析

本题考查分布式数据库基本概念。

分片透明是指用户或应用程序不需要知道逻辑上访问的表具体是怎么分块存储的。复制透明是指采用复制技术的分布方法，用户不需要知道数据是复制到哪些节点，如何复制的。位置透明是指用户无须知道数据存放的物理位置，逻辑透明，即局部数据模型透明，是指用户或应用程序无须知道局部场地使用的是哪种数据模型。

参考答案

（53）D　（54）A

试题（55）、（56）

设有关系模式R（A_1，A_2，A_3，A_4，A_5，A_6），其中：函数依赖集F＝{$A_1 \to A_2$，$A_1A_3 \to A_4$，$A_5A_6 \to A_1$，$A_2A_5 \to A_6$，$A_3A_5 \to A_6$}，则 （55） 是关系模式R的一个主键，R规范化程度最高达到 （56） 。

（55）A. A_1A_4　B. A_2A_4　C. A_3A_5　D. A_4A_5

（56）A. 1NF　B. 2NF　C. 3NF　D. BCNF

试题（55）、（56）分析

本题主要考核关系模式规范化方面的相关知识。

试题（55）的正确答案为C。因为根据函数依赖集F可知属性A_3和A_5只出现在函数依赖的左部，故必为候选关键字属性，又因为A_3A_5可以决定关系R中的全部属性，故关系模式R的一个主键是A_3A_5。

试题（56）的正确答案为B。因为根据函数依赖集F可知，R中的每个非主属性完全函数依赖于A_3A_5，但该函数依赖集中存在传递依赖，所以R是2NF。

参考答案

（55）C　（56）B

试题（57）

对于一个长度为n（n>1）且元素互异的序列，令其所有元素依次通过一个初始为空的栈后，再通过一个初始为空的队列。假设队列和栈的容量都足够大，且只要栈非空就可以进行出栈操作，只要队列非空就可以进行出队操作，那么以下叙述中，正确的是 （57） 。

（57）A. 出队序列和出栈序列一定互为逆序

B. 出队序列和出栈序列一定相同

C. 入栈序列与入队序列一定相同

D. 入栈序列与入队序列一定互为逆序

试题（57）分析

本题考查数据结构基础知识。

栈和队列都是线性的数据结构。栈的操作要求是入栈和出栈都在表尾进行，即在栈中有多个元素时，后进去的元素先出来，特点是后进先出，元素入栈的顺序与出栈的顺序可以相同也可以不同。而队列的修改要求是在队尾加入元素，在队头删除元素，特点是先进先出，元素的入队顺序与出队顺序一定相同。

将一个栈和队列连接后，进出队列的元素顺序是相同的，而进入队列的元素顺序正是从栈中出来的元素顺序，因此，正确的叙述为出队序列与出栈序列一定相同。

参考答案

（57）B

试题（58）

设某 n 阶三对角矩阵 $A_{n\times n}$ 的示意图如下图所示。若将该三对角矩阵的非零元素按行存储在一维数组 B[k]($1\le k\le 3\times n-2$)中，则 k 与 i、j 的对应关系是＿（58）＿。

$$A_{n\times n}=\begin{bmatrix} a_{1,1} & a_{1,2} & & & & 0 \\ a_{2,1} & a_{2,2} & a_{2,3} & & & \\ & a_{3,2} & a_{3,3} & a_{3,4} & & \\ & & \cdots & \cdots & \cdots & \\ & 0 & & \cdots & \cdots & \cdots \\ & & & & a_{n,n-1} & a_{n,n} \end{bmatrix}$$

（58）A．$k = 2i+j-2$　　B．$k = 2i-j+2$

C．$k = 3i+j-1$　　D．$k = 3i-j+2$

试题（58）分析

本题考查数据结构基础知识。

解答该问题需先计算排列在 $A_{i,j}$ 之前的元素个数。

在按行存储方式下，存储在 $A_{i,j}$ 之前的元素分为 i-1 行，除第 1 行外，每行 3 个元素。在第 i 行上，$A_{i,j}$ 之前的元素个数分为三种情况：i>j 时为 0 个，i=j 时有 1 个，i<j 时为 2 个，概括为 j-i+1 个。

综上，排列在 $A_{i,j}$ 之前的元素个数为 $(i-1)\times 3-1+j-i+1$，即 $2i+j-3$。

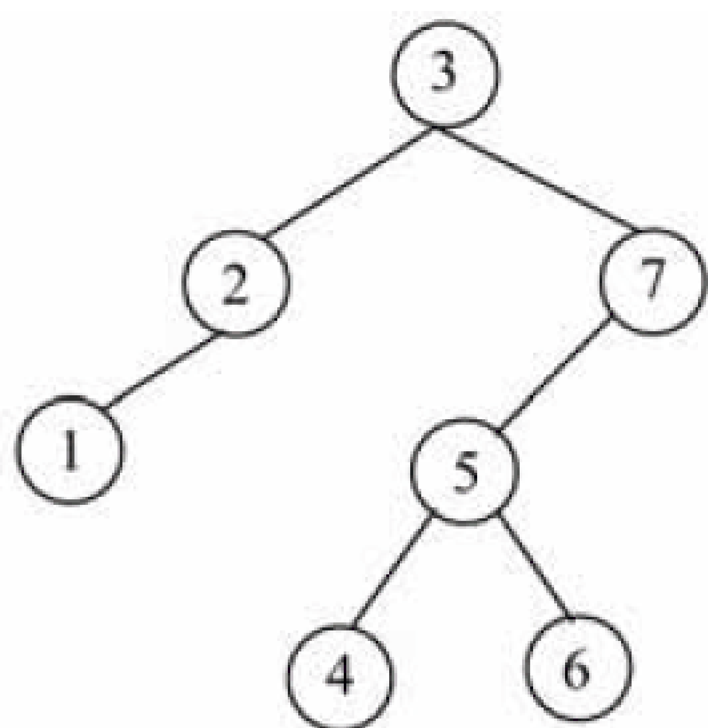
由于数组 B 的下标从 1 开始，所以 $k = 2i+j-3+1$。

参考答案

（58）A

试题（59）

对于非空的二叉树，设 D 代表根结点，L 代表根结点的左子树，R 代表根结点的右子树。若对下图所示的二叉树进行遍历后的结点序列为 7 6 5 4 3 2 1，则遍历方式是___（59）___。



（59）A．LRD　　B．DRL　　C．RLD　　D．RDL

试题（59）分析

本题考查数据结构基础知识。

由于序列的第一个元素是结点 7，最后一个元素是结点 1，因此，左右子树的遍历顺序是先右后左。观察结点 7 的左子树，遍历顺序为 654，因此是中序遍历过程。所以答案为 RDL。

参考答案

（59）D

试题（60）

在 55 个互异元素构成的有序表 A[1..55]中进行折半查找（或二分查找，向下取整）。若需查找的元素等于 A[19]，则在查找过程中参与比较的元素依次为___（60）___、A[19]。

（60）A．A[28]、A[30]、A[15]、A[20]

B．A[28]、A[14]、A[21]、A[17]

C．A[28]、A[15]、A[22]、A[18]

D．A[28]、A[18]、A[22]、A[20]

试题（60）分析

本题考查数据结构基础知识。

对 55 个元素构成的有序表进行折半查找时，可用判定树描述查找过程，由于 A[19]小于中间元素 A[28]，所以判定树的左分支如下图所示。从中可知，查找过程中参与比较的元素分别为 A[28]、A[14]、A[21]、A[17]、A[19]。

参考答案

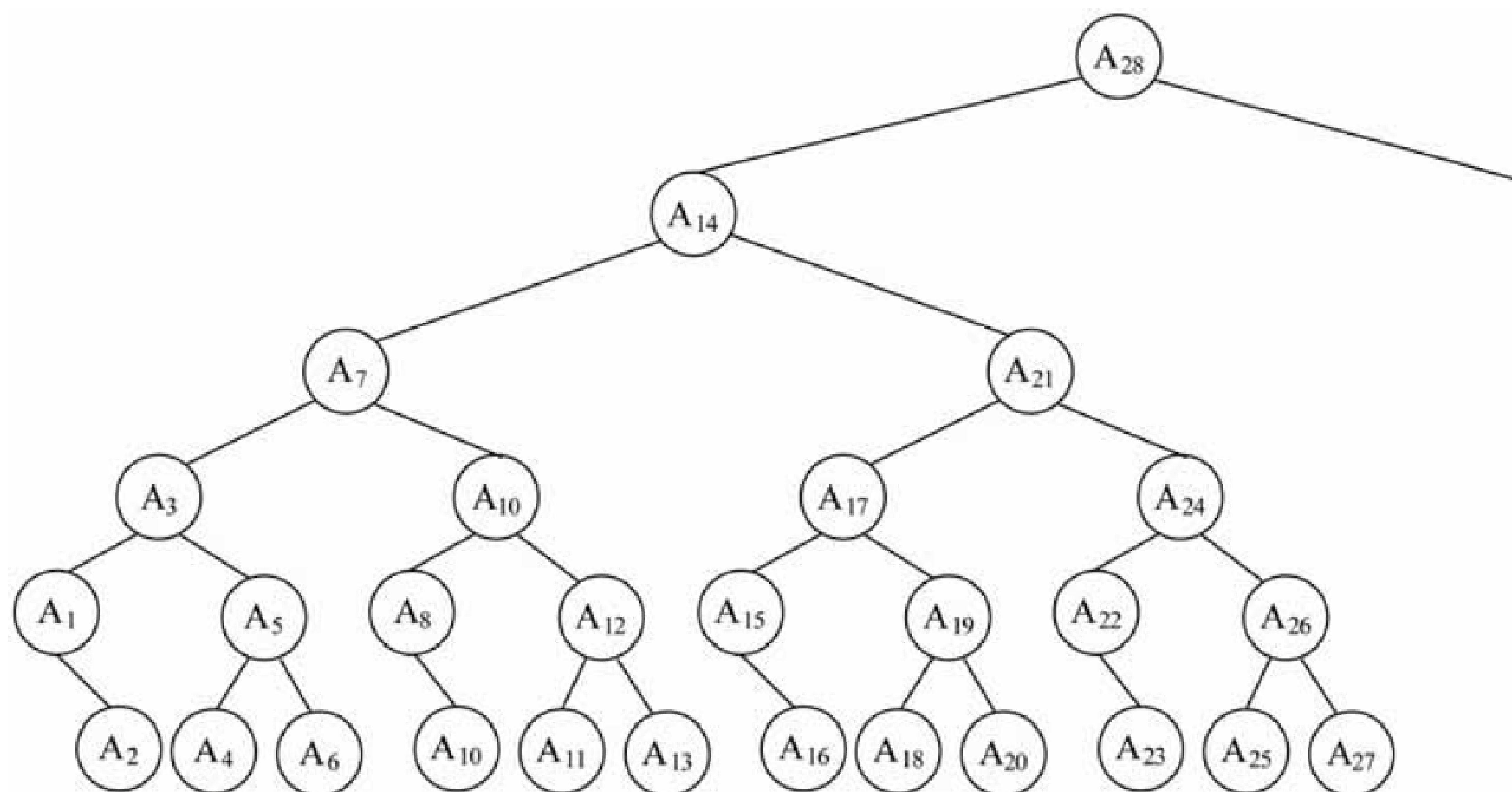
（60）B

试题（61）

设一个包含 n 个顶点、e 条弧的简单有向图采用邻接矩阵存储结构（即矩阵元素

A[i][j]等于 1 或 0，分别表示顶点 i 与顶点 j 之间有弧或无弧），则该矩阵的非零元素数目为__（61）__。

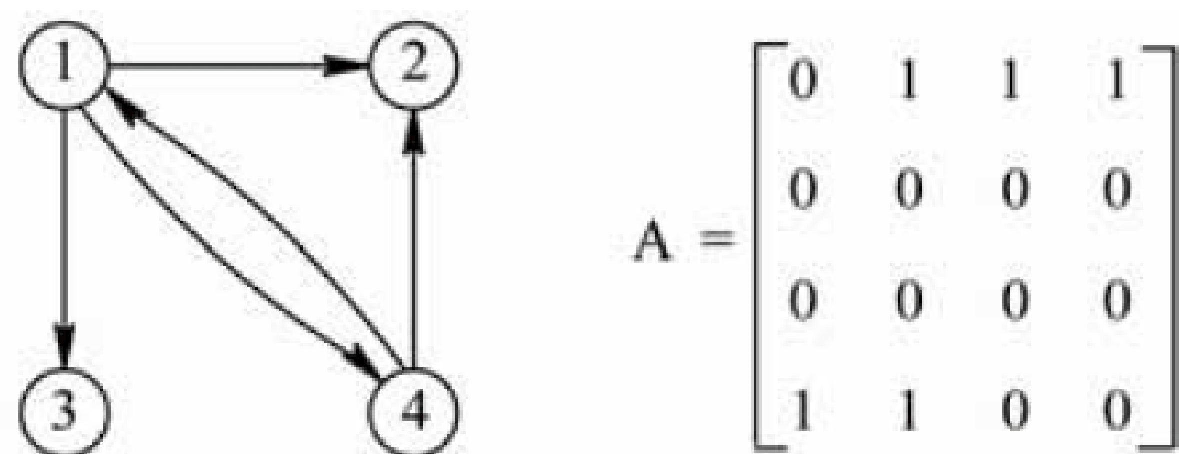
（61）A. e　　B. 2e　　C. n-e　　D. n+e



试题（61）分析

本题考查数据结构基础知识。

通过一个例子说明。某有向图及其邻接矩阵如下图所示。



邻接矩阵中的每个非零元素都表示一条弧，所以非零元素数目为弧的个数 e。

参考答案

（61）A

试题（62）、（63）

已知算法 A 的运行时间函数为 $T(n) = 8T(n/2) + n^2$，其中 n 表示问题的规模，则该算法的时间复杂度为__（62）__。另已知算法 B 的运行时间函数为 $T(n) = XT(n/4) + n^2$，其中 n 表示问题的规模。对充分大的 n，若要算法 B 比算法 A 快，则 X 的最大值为__（63）__。

（62）A. $\Theta(n)$　　B. $\Theta(n\lg n)$　　C. $\Theta(n^2)$　　D. $\Theta(n^3)$

（63）A. 15　　B. 17　　C. 63　　D. 65

试题（62）、（63）分析

本题考查算法分析的基础知识。

根据主方法，先计算算法 A 的时间复杂度，a=8,b=2,$\log_b a = \log_2 8 = 3$，而 $f(n)=n^2$，

因此时间复杂度为 $\Theta(n^3)$。然后计算算法 B 的时间复杂度，a=X,b=4, $\log_b a = \log_4 X$，而 f(n)=n^2，若算法 B 和算法 A 的效率一样，则 X 应该为 64（$\log_4 64 = 3$），而现在要使得 B 比 A 快，则 X 应该比 64 小，因此最大的整数应该为 63。

参考答案

（62）D （63）C

试题（64）、（65）

在某应用中，需要先排序一组大规模的记录，其关键字为整数。若这组记录的关键字基本上有序，则适宜采用__（64）__排序算法。若这组记录的关键字的取值均在 0 到 9 之间（含），则适宜采用__（65）__排序算法。

（64）A. 插入 B. 归并 C. 快速 D. 计数

（65）A. 插入 B. 归并 C. 快速 D. 计数

试题（64）、（65）分析

本题考查算法设计和排序的基础知识。

排序是一类最基本的操作，因此要求考生熟悉一些典型的排序算法，包括其算法思想、时空复杂度以及应用场合。若数据基本有序，插入排序应该是最佳选择，输入数据是否有序对归并和计数排序算法并没有影响。对传统的快速排序算法，输入数据有序反而使其效率最低。若关键字取值范围较小，则计数排序是最佳选择，因为在该情况下，该算法的时间复杂度为线性时间。

参考答案

（64）A （65）D

试题（66）

集线器与网桥的区别是：__（66）__。

（66）A. 集线器不能检测发送冲突，而网桥可以检测冲突

B. 集线器是物理层设备，而网桥是数据链路层设备

C. 网桥只有两个端口，而集线器是一种多端口网桥

D. 网桥是物理层设备，而集线器是数据链路层设备

试题（66）分析

集线器是物理层设备，相当于在 10BASE2 局域网中把连接工作站的同轴电缆收拢在一个盒子里，这个盒子只起到接收和发送的功能，可以检测发送冲突，但不能识别数据链路层的帧。网桥是数据链路层设备，它可以识别数据链路层 MAC 地址，有选择地把帧发送到输出端口，网桥也可以有多个端口，如果网桥端口很多，并配置了加快转发的硬件，就成为局域网交换机。

参考答案

（66）B

试题（67）

POP3 协议采用__（67）__模式，客户端代理与 POP3 服务器通过建立 TCP 连接来传送数据。

（67）A．Browser/Server　B．Client/Server　C．Peer to Peer　D．Peer to Server

试题（67）分析

本题考查 POP3 协议及 POP3 服务器方面的基础知识。

POP3 协议是 TCP/IP 协议簇中用于邮件接收的协议。邮件客户端通过与服务器之间建立 TCP 连接，采用 Client/Server 计算模式来传送邮件。

参考答案

（67）B

试题（68）

TCP 使用的流量控制协议是__（68）__。

（68）A．固定大小的滑动窗口协议　B．后退 N 帧的 ARQ 协议
C．可变大小的滑动窗口协议　D．停等协议

试题（68）分析

TCP 的流量控制采用了可变大小的滑动窗口协议，由接收方指明接收缓冲区的大小（字节数），发送方发送了规定的字节数后等待接收方的下一次请求。固定大小的滑动窗口协议用在数据链路层的 HDLC 中。可变大小的滑动窗口协议可以应付长距离通信过程中线路延迟不确定的情况，而固定大小的滑动窗口协议则适合链路两端点之间通信延迟固定的情况。

参考答案

（68）C

试题（69）

以下 4 种路由中，__（69）__路由的子网掩码是 255.255.255.255。

（69）A．远程网络　B．静态　C．默认　D．主机

试题（69）分析

主机路由的子网掩码是 255.255.255.255。网络路由要指明一个子网，所以不可能为全 1，默认路由是访问默认网关，而默认网关与本地主机属于同一个子网，其子网掩码也应该与网络路由相同，对静态路由也是同样的道理。

参考答案

（69）D

试题（70）

以下关于层次化局域网模型中核心层的叙述，正确的是__（70）__。

（70）A．为了保障安全性，对分组要进行有效性检查
B．将分组从一个区域高速地转发到另一个区域

C．由多台二、三层交换机组成

D．提供多条路径来缓解通信瓶颈

试题（70）分析

在层次化局域网模型中，核心层的主要功能是将分组从一个区域高速地转发到另一个区域。核心层是因特网络的高速骨干，由于其重要性，因此在设计中应该采用冗余组件设计，使其具备高可靠性，能快速适应变化。在设计核心层设备的功能时，应尽量避免使用数据包过滤、策略路由等降低数据包转发处理的特性，以优化核心层获得低延迟和良好的可管理性。

汇聚层是核心层和接入层的分界点，应尽量将资源访问控制、核心层流量的控制等都在汇聚层实施。汇聚层应向核心层隐藏接入层的详细信息，汇聚层向核心层路由器进行路由宣告时，仅宣告多个子网地址汇聚而形成的一个网络。另外，汇聚层也会对接入层屏蔽网络其他部分的信息，汇聚层路由器可以不向接入路由器宣告其他网络部分的路由，而仅仅向接入设备宣告自己为默认路由。

接入层为用户提供了在本地网段访问应用系统的能力，接入层要解决相邻用户之间的互访需要，并且为这些访问提供足够的带宽。接入层还应该适当负责一些用户管理功能，包括地址认证、用户认证和计费管理等内容。接入层还负责一些用户信息收集工作，例如用户的 IP 地址、MAC 地址和访问日志等信息。

参考答案

（70）B

试题（71）～（75）

In a world where it seems we already have too much to do, and too many things to think about, it seems the last thing we need is something new that we have to learn.

But use cases do solve a problem with requirements: with ___(71)___ declarative requirements it's hard to describe steps and sequences of events.

Use cases, stated simply, allow description of sequences of events that, taken together, lead to a system doing something useful. As simple as this sounds, this is important. When confronted only with a pile of requirements, it's often ___(72)___ to make sense of what the authors of the requirements really wanted the system to do. In the preceding example, use cases reduce the ambiguity of the requirements by specifying exactly when and under what conditions certain behavior occurs; as such, the sequence of the behaviors can be regarded as a requirement. Use cases are particularly well suited to capturing these kind of requirements. Although this may sound simple, the fact is that ___(73)___ requirement capture approaches, with their emphasis on declarative requirements and "shall" statements, completely fail to capture the ___(74)___ of the system's behavior. Use cases are a simple yet powerful way to express the behavior of the system in way that all stakeholders can easily

understand.

But, like anything, use cases come with their own problems, and as useful as they are, they can be __(75)__. The result is something that is as bad, if not worse, than the original problem. Therein it's important to utilize use cases effectively without creating a greater problem than the one you started with.

(71) A. plenty　B. loose　C. extra　D. strict
(72) A. impossible　B. possible　C. sensible　D. practical
(73) A. modern　B. conventional　C. different　D. formal
(74) A. statics　B. nature　C. dynamics　D. originals
(75) A. misapplied　B. applied　C. used　D. powerful

参考译文

在这个世界上，似乎我们有太多的事情要去做，有太多的事情要去思考，那么需要做的最后一件事就是必须学习新事物。

而用例恰恰可以解决带有需求的问题：如果具有严格声明的需求，则很难描述事件的步骤和序列。

简单地说，用例可以将事件序列的说明放在一起，引导系统完成有用的任务。正如听起来一样简单——这很重要。在面对很多需求的时候，通常不太可能理解需求的作者真正想要系统做什么。在前面的例子中，通过指定特定行为发生的时间和条件，用例减少了需求的不确定性。这样的话，行为的顺序就可以当作是一种需求。用例特别适用于捕捉这类需求。尽管听起来可能很简单，但事实情况是由于常规的需求捕捉方法所侧重的是声明需求和“应该怎么样”的陈述，因此完全无法捕捉系统行为的动态方面。用例是一种简单而有效的表达系统行为的方式，使用这种方式所有参与者都很容易理解。

但是与任何事物一样，用例也存在自己的问题——在用例非常有用的同时，人们也可能误用它，结果就产生了比原来更为糟糕的问题。因此重点在于：如何有效地使用用例，而又不会产生出比原来更严重的问题。

参考答案

(71) D　(72) A　(73) B　(74) C　(75) A

第 28 章　2015 下半年软件设计师下午试题分析与解答

试题一（共 15 分）

阅读下列说明和图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

【说明】

某慕课教育平台欲添加在线作业批改系统，以实现高效的作业提交与批改，并进行统计。学生和讲师的基本信息已经初始化为数据库中的学生表和讲师表。系统的主要功能如下。

（1）提交作业。验证学生标识后，学生将电子作业通过在线方式提交，并进行存储。系统给学生发送通知表明提交成功，通知中包含唯一编号；并通知讲师有作业提交。

（2）下载未批改作业。验证讲师标识后，讲师从系统中下载学生提交的作业。下载的作业将显示在屏幕上。

（3）批改作业。讲师按格式为每个题目进行批改打分，并进行整体评价。

（4）上传批改后的作业。将批改后的作业（包括分数和评价）返回给系统，进行存储。

（5）记录分数和评价。将批改后的作业的分数和评价记录在学生信息中，并通知学生作业已批改。

（6）获取已批改作业。根据学生标识，给学生查看批改后的作业，包括提交的作业、分数和评价。

（7）作业抽检。根据教务人员标识抽取批改后的作业样本，给出抽检意见，然后形成抽检报告给讲师。

现采用结构化方法对在线作业批改系统进行分析与设计，获得如图 1-1 所示的上下文数据流图和图 1-2 所示的 0 层数据流图。

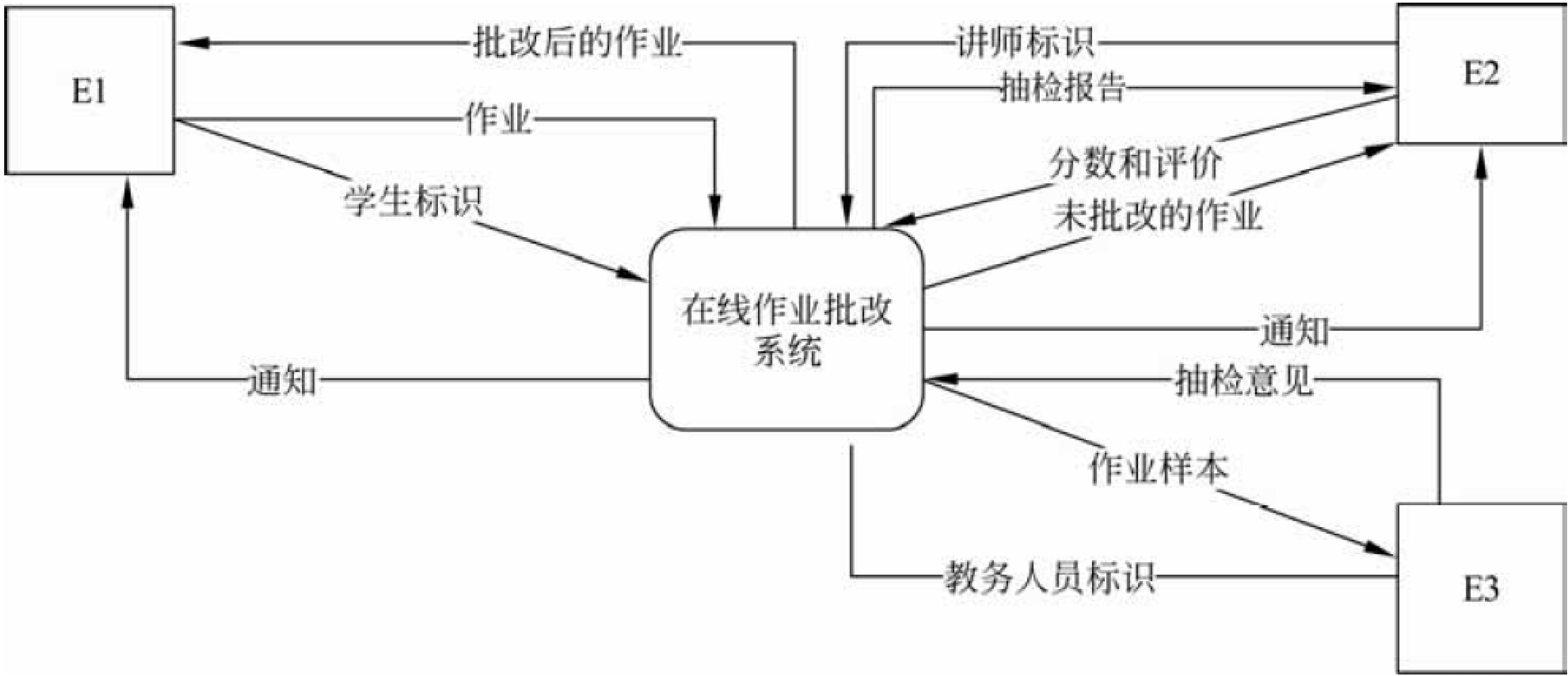


图 1-1　上下文数据流图

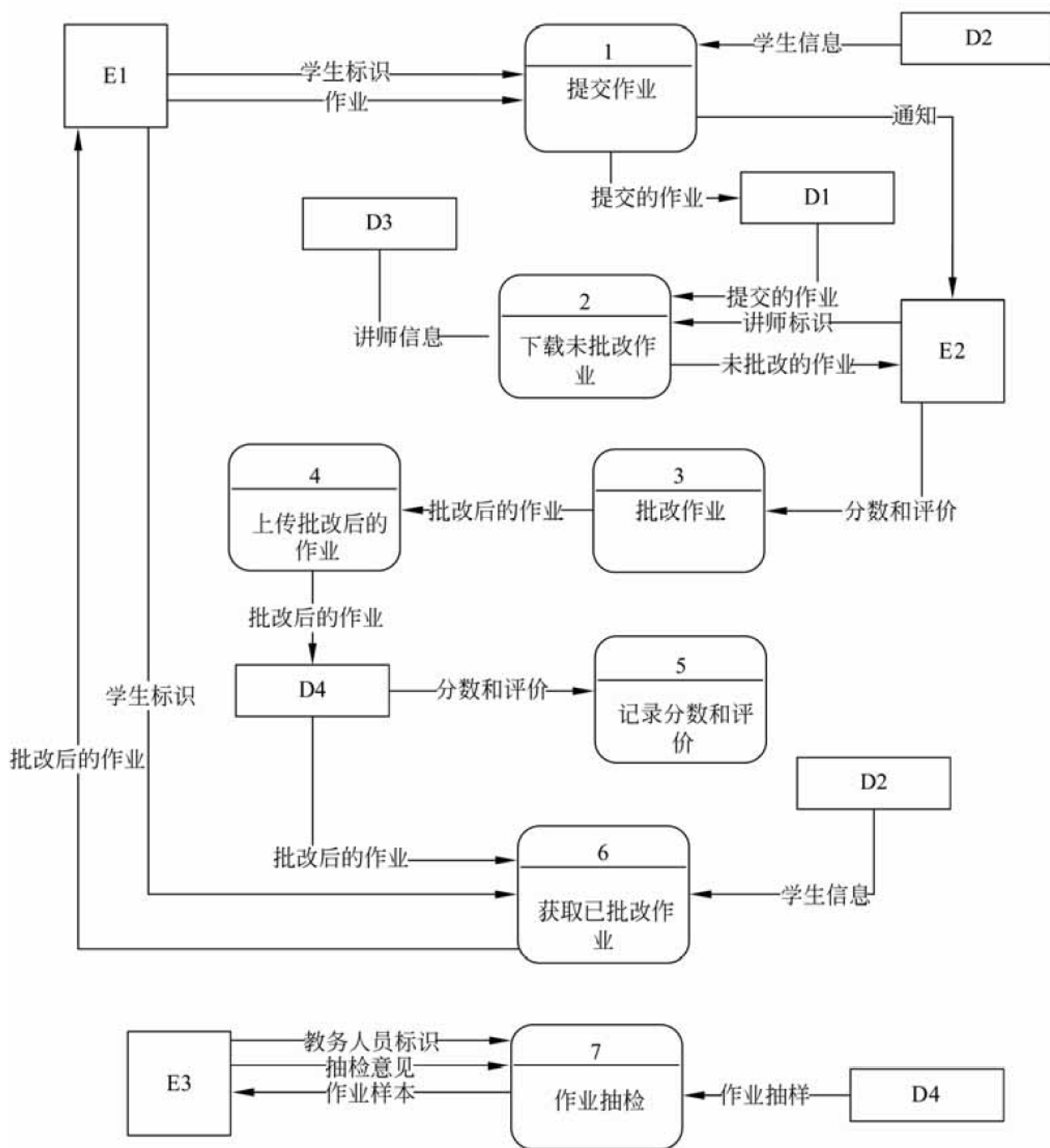


图 1-2　0 层数据流图

【问题 1】（3 分）

使用说明中的词语，给出图 1-1 中的实体 E1～E3 的名称。

【问题 2】（4 分）

使用说明中的词语，给出图 1-2 中的数据存储 D1～D4 的名称。

【问题 3】（6 分）

根据说明和图中术语，补充图 1-2 中缺失的数据流及其起点和终点。

【问题 4】（2 分）

若发送给学生和讲师的通知是通过第三方 Email 系统进行的，则需要对图 1-1 和图 1-2 进行哪些修改？用 100 字以内文字加以说明。

试题一分析

本题考查采用结构化方法进行系统分析与设计，主要考查数据流图（DFD）的应用，是比较传统的题目，考点与往年类似，要求考生细心分析题目中所描述的内容。

DFD 是一种便于用户理解、分析系统数据流程的图形化建模工具，是系统逻辑模型的重要组成部分。上下文 DFD（顶层 DFD）通常用来确定系统边界，将待开发系统看作一个大的加工（处理），然后根据系统从哪些外部实体接收数据流，以及系统将数据流发送到哪些外部实体，建模出的上下文数据流图中只有唯一的一个加工和一些外部实体，以及这两者之间的输入输出数据流。0 层 DFD 在上下文确定的系统外部实体以及与外部实体的输入输出数据流的基础上，将上下文 DFD 中的加工分解成多个加工，识别这些加工的输入输出数据流，使得所有上下文 DFD 中的输入数据流经过这些加工之后变换成上下文 DFD 的输出数据流。根据 0 层 DFD 中加工的复杂程度进一步建模加工的内容。

在建分层 DFD 时，根据需求情况可以将数据存储建模在不同层次的 DFD 中，注意，在绘制下层数据流图时要保持父图与子图平衡。父图中某加工的输入输出数据流必须与其子图的输入输出数据流在数量和名字上相同，或者父图中的一个输入（或输出）数据流对应于子图中几个输入（或输出）数据流，而子图中组成这些数据流的数据项的全体正好是父图中的这一个数据流。

【问题 1】

本问题考查上下文 DFD，要求确定外部实体。通过考查系统的主要功能不难发现，系统中涉及到学生、讲师和教务人员，没有提到其他与系统交互的外部实体。根据描述（1）中“学生将电子作业通过在线的方式提交”，（2）中“讲师从系统中下载学生提交的作业”，（7）中“根据教务人员标识抽取批改后的作业样本，给出抽检意见”等信息，从而即可确定 E1 为“学生”实体，E2 为“讲师”实体，E3 为“教务人员”实体。

【问题 2】

本问题要求确定 0 层数据流图中的数据存储。分析说明中和数据存储有关的描述，说明（1）中“验证学生标识后，学生将电子作业通过在线的方式提交，并进行存储”，说明（2）中“讲师从系统中下载学生提交的作业”，可知 D1 为提交的作业表；说明（2）中“验证讲师标识后”，可知 D3 为讲师表；说明（4）中 “将批改后的作业（包括分数和评价）返回给系统，进行存储”，可知 D4 为批改后的作业表。

【问题 3】

本问题要求补充缺失的数据流及其起点和终点。对照图 1-1 和图 1-2 的输入、输出数据流，数量不同，考查图 1-1 中输出至 E2 的数据流，有“通知”和“抽检报告”，而图 1-2 中缺少了这几条数据流，所以需要确定这几条数据流或者其分解的数据流的起点

或终点。

下面考查说明中的功能。先考查“通知”，功能（1）中“系统给学生发送通知表明提交成功”，对照图 1-2，加工 1 没有到实体 E1 学生的“通知”数据流；功能（5）中“并通知学生作业已批改”，对照图 1-2，加工 5 没有到实体 E1 学生的数据流“通知”。进一步加以区别，加工 1 到实体 E1 学生的数据流为“提交成功通知”，加工 5 到实体 E1 学生缺少的数据流应为“作业已批改通知”。这两条数据流是上下文数据流图中对数据流“通知”的分解。再根据功能（7）中“然后形成抽检报告给讲师”，对照图 1-2 中加工 7 应该有数据流“抽检报告”，终点为 E2 讲师实体。

下面再仔细核对说明和图 1-2 之间是否还有遗失的数据流。不难发现，功能（3）中“将批改后的作业的分数和评价记录在学生信息中”，而图 1-2 中加工 5 从 D4 批改后的作业表中读取了分数和评价，并没有存入学生表，所以，此处遗失了数据流“分数与评价”，起点是加工 5，终点是 D2 学生表。

【问题 4】

DFD 中，外部实体可以是用户，也可以是其他交互的系统。如果某功能交互的是外部系统，本题中是通过第三方 Email 系统，即系统需要将发送给学生和教师的通知相关信息发送给第三方 Email 系统。然后由第三方 Email 系统给学生和教师发送邮件，此时第三方 Email 系统即为外部实体，而非本系统内部加工，因此需要对图 1-1 和图 1-2 进行修改，添加外部实体“Email 系统”，并将数据流通知的终点都改为 Email 系统。在图 1-1 中将唯一加工到 E1 和 E2 的通知数据流终点改为“Email 系统”。在图 1-2 中，除了将加工 1 到 E2 的数据流通知的终点改为“Email 系统”，还需要将【问题 3】补充“提交成功通知”和“作业已批改通知”的终点也改为“Email 系统”。

试题一参考答案

【问题 1】

E1：学生　　E2：讲师　　E3：教务人员

【问题 2】

D1：提交的作业表　　D2：学生表　　D3：讲师表　　D4：批改后的作业表

【问题 3】

数　据　流	起　　点	终　　点
提交成功通知	1 或 提交作业	E1 或 学生
作业已批改通知	5 或 记录分数和评价	E1 或 学生
分数和评价	5 或 记录分数和评价	D2 或 学生表
抽检报告	7 或 作业抽检	E2 或 讲师

【问题 4】

将 Email 系统作为外部实体，并将通知的终点全部改为 Email 系统。

试题二（共 15 分）

阅读下列说明，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某企业拟构建一个高效、低成本、符合企业实际发展需要的办公自动化系统。工程师小李主要承担该系统的公告管理和消息管理模块的研发工作。公告管理模块的主要功能包括添加、修改、删除和查看公告。消息管理模块的主要功能是消息群发。

小李根据前期调研和需求分析进行了概念模型设计，具体情况分述如下。

【需求分析结果】

（1）该企业设有研发部、财务部、销售部等多个部门，每个部门只有一名部门经理，有多名员工，每名员工只属于一个部门。部门信息包括：部门号、名称、部门经理和电话，其中部门号唯一确定部门关系的每一个元组。

（2）员工信息包括：员工号、姓名、岗位、电话和密码。员工号唯一确定员工关系的每一个元组；岗位主要有经理、部门经理、管理员等，不同岗位具有不同的权限。一名员工只对应一个岗位，但一个岗位可对应多名员工。

（3）消息信息包括：编号、内容、消息类型、接收人、接收时间、发送时间和发送人。其中（编号，接收人）唯一标识消息关系中的每一个元组。一条消息可以发送给多个接收人，一个接收人可以接收多条消息。

（4）公告信息包括：编号、标题、名称、内容、发布部门、发布时间。其中编号唯一确定公告关系的每一个元组。一份公告对应一个发布部门，但一个部门可以发布多份公告；一份公告可以被多名员工阅读，一名员工可以阅读多份公告。

【概念模型设计】

根据需求分析阶段收集的信息，设计的实体联系图（不完整）如图 2-1 所示。

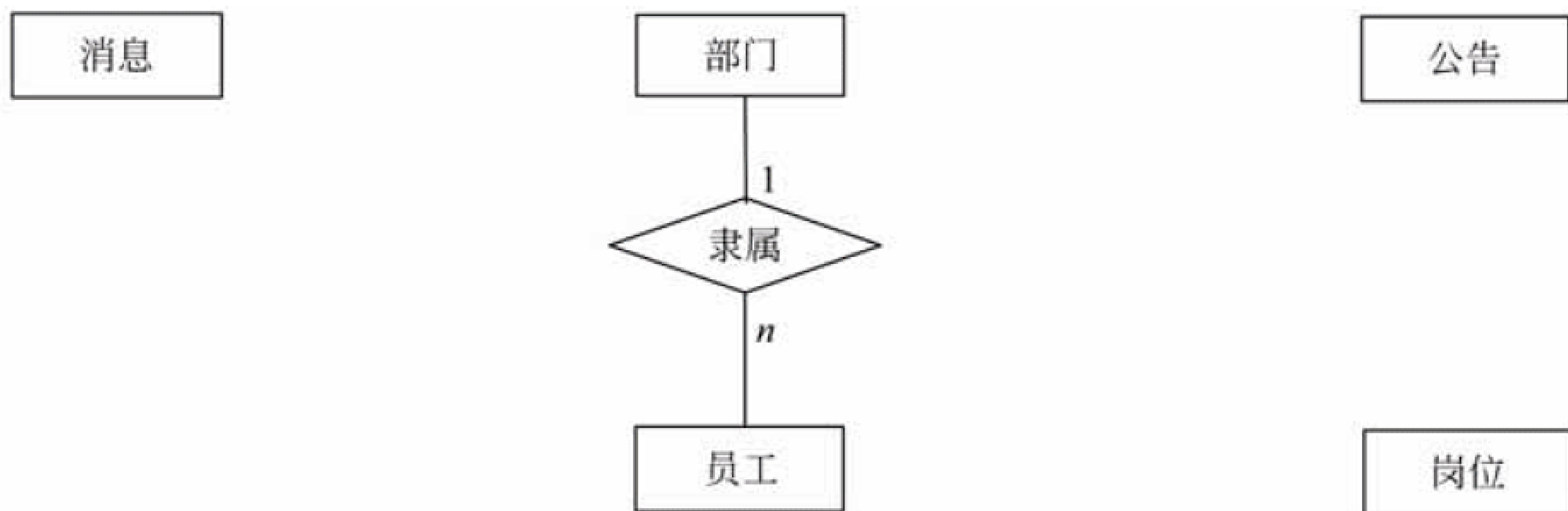


图 2-1　实体联系图

【逻辑结构设计】

根据概念模型设计阶段完成的实体联系图，得出如下关系模式（不完整）：

部门（________（a）________，部门经理，电话）

员工（员工号，姓名，岗位号，部门号，电话，密码）

岗位（岗位号，名称，权限）

消息（______(b)______，消息类型，接收时间，发送时间，发送人）

公告（______(c)______，名称，内容，发布部门，发布时间）

阅读公告（______(d)______，阅读时间）

【问题 1】（5 分）

根据问题描述，补充四个联系，完善图 2-1 所示的实体联系图。联系名可用联系 1、联系 2、联系 3 和联系 4 代替，联系的类型分为 1:1、1:n 和 m:n（或 1:1、1:*和*:*）。

【问题 2】（8 分）

（1）根据实体联系图，将关系模式中的空（a）～（d）补充完整。

（2）给出“消息”和“阅读公告”关系模式的主键与外键。

【问题 3】（2 分）

消息和公告关系中都有“编号”属性，请问它是属于命名冲突吗？用 100 字以内文字说明原因。

试题二分析

本题考查数据库概念结构设计及概念结构向逻辑结构转换的过程。

此类题目要求考生认真阅读题目对现实问题的描述，经过分类、聚集、概括等方法，从中确定实体及其联系。题目已经给出了 4 个实体，需要根据需求描述，给出实体间的联系。

【问题 1】

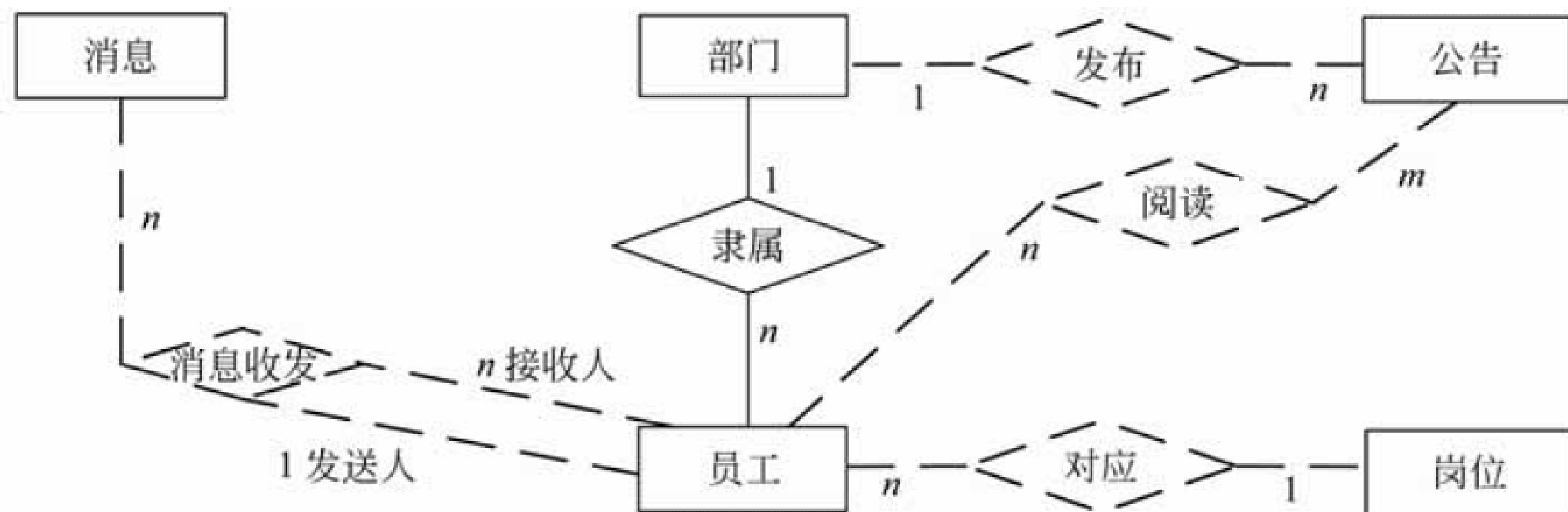
根据题意“一个员工只对应一个岗位，但一个岗位可对应多名员工”，可以得出员工与岗位之间的对应联系类型为 n:1。

由“一条消息可以发送给多个接收人，一个接收人可以接收多条消息”，可以得出员工与消息之间的收发联系类型为 1:n:m。

由“一份公告对应一个发布部门，但一个部门可以发布多份公告”可以得出部门与公告间的所属联系类型为 1:n。

由“一份公告可以有多个员工阅读，一个员工可以阅读多份公告”，可以得出，公告与员工之间的阅读联系类型为 n:m。

完整的 ER 图如下：



【问题 2】

（1）根据题意，完整的关系模式如下：

部门（部门号，名称，部门经理，电话）

员工（员工号，姓名，岗位号，部门号，电话，密码）

岗位（岗位号，名称，权限）

消息（编号，内容，接收人，消息类型，接收时间，发送时间，发送人）

公告（编号，标题，名称，内容，发布部门，发布时间）

阅读公告（公告编号，员工号，阅读时间）

（2）消息关系模式和阅读公告关系模式的主键和外键的分析如下：

根据题意，消息关系模式的主键为（编号，接收人）。由于接收人、发送人都应参考员工关系的员工号，因此接收人、发送人为消息关系的外键。

根据题意，阅读公告关系模式的主键为（公告编号，员工号）。外键为公告编号、员工号，因为公告编号应参考公告关系的编号，而编号是公告关系的主键，所以公告编号是阅读公告关系的外键；又因为员工号应参考员工关系的员工号，而员工号是员工关系的主键，所以公告关系的员工号为外键。

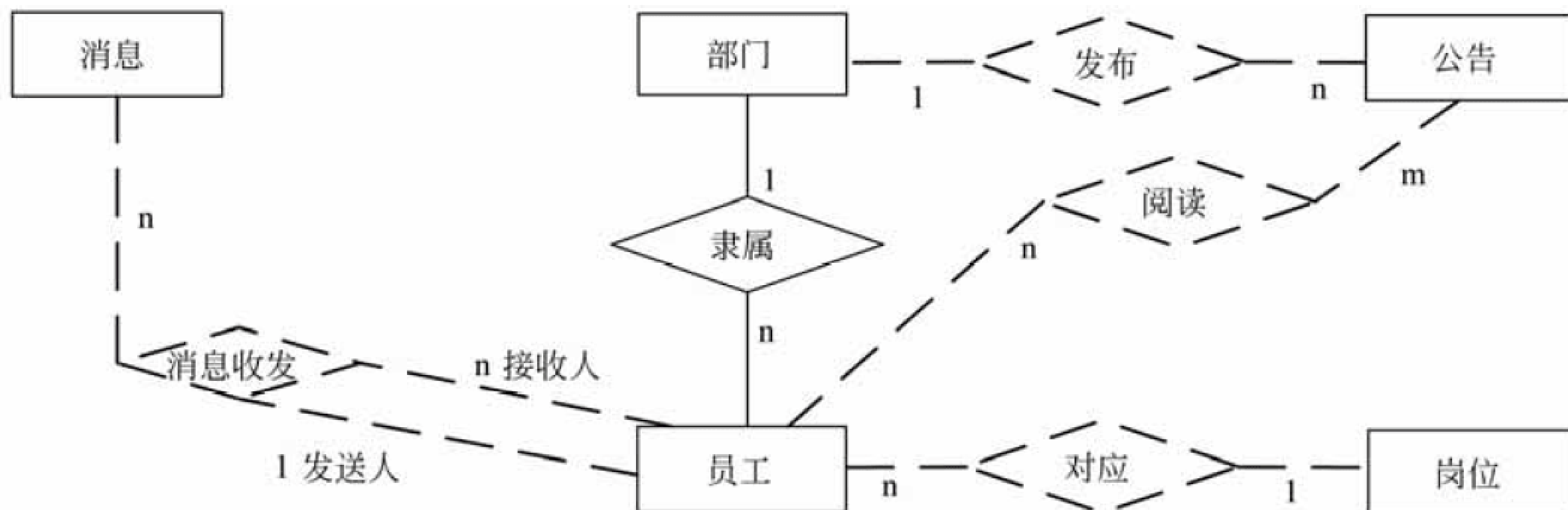
【问题 3】

消息和公告关系中都有“编号”属性，但是它们不属于命名冲突。因为这两个属性分别属于两个不同的关系模式，可以通过“关系名.属性名”区别，即可以用“消息.编号”和“公告.编号”来区别。

试题二参考答案

【问题 1】

联系名称可不做要求，但不能出现重名。



【问题 2】

（1）（a）部门号，名称

（b）编号，内容，接收人

（c）编号，标题

（d）公告编号，员工号（注：编号，员工号也正确）

（2）消息关系模式的主键：编号，接收人

外键：接收人、发送人

阅读公告关系模式的主键：公告编号，员工号

外键：公告编号，员工号

【问题 3】

不属于命名冲突。因为这两个属性分别属于两个不同的关系模式，可以通过“关系名.属性名”区别，即可以用“消息.编号”和“公告.编号”来区别。

试题三（15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某出版社拟开发一个在线销售各种学术出版物的网上商店（ACShop），其主要的功能需求描述如下：

（1）ACShop 在线销售的学术出版物包括论文、学术报告或讲座资料等。

（2）ACShop 的客户分为两种：未注册客户和注册客户。

（3）未注册客户可以浏览或检索出版物，将出版物添加到购物车中。未注册客户进行注册操作之后，成为 ACShop 注册客户。

（4）注册客户登录之后，可将待购买的出版物添加到购物车中，并进行结账操作。结账操作的具体流程描述如下：

① 从预先填写的地址列表中选择一个作为本次交易的收货地址。如果没有地址信息，可以添加新地址信息。

② 选择付款方式。ACShop 支持信用卡付款或银行转账两种方式。注册客户可以从预先填写的信用卡或银行账号中选择一个付款，若没有付款方式信息，则可以添加新付款方式。

③ 确认提交购物车中待购买的出版物，ACShop 会自动生成与之相对应的订单。

（5）管理员负责维护在线销售的出版物目录，包括添加新出版物或者更新在售出版物信息等操作。

现采用面向对象方法分析并设计该网上商店 ACShop，得到如图 3-1 所示的用例图和图 3-2 所示的类图。

【问题 1】（4 分）

根据说明中的描述，给出图 3-1 中（1）～（4）所对应的用例名。

【问题 2】（4 分）

根据说明中的描述，分别说明用例“添加新地址”和“添加新付款方式”会在何种情况下由图 3-1 中的用例（3）和（4）扩展而来？

【问题 3】（7 分）

根据说明中的描述，给出图 3-2 中（1）～（7）所对应的类名。

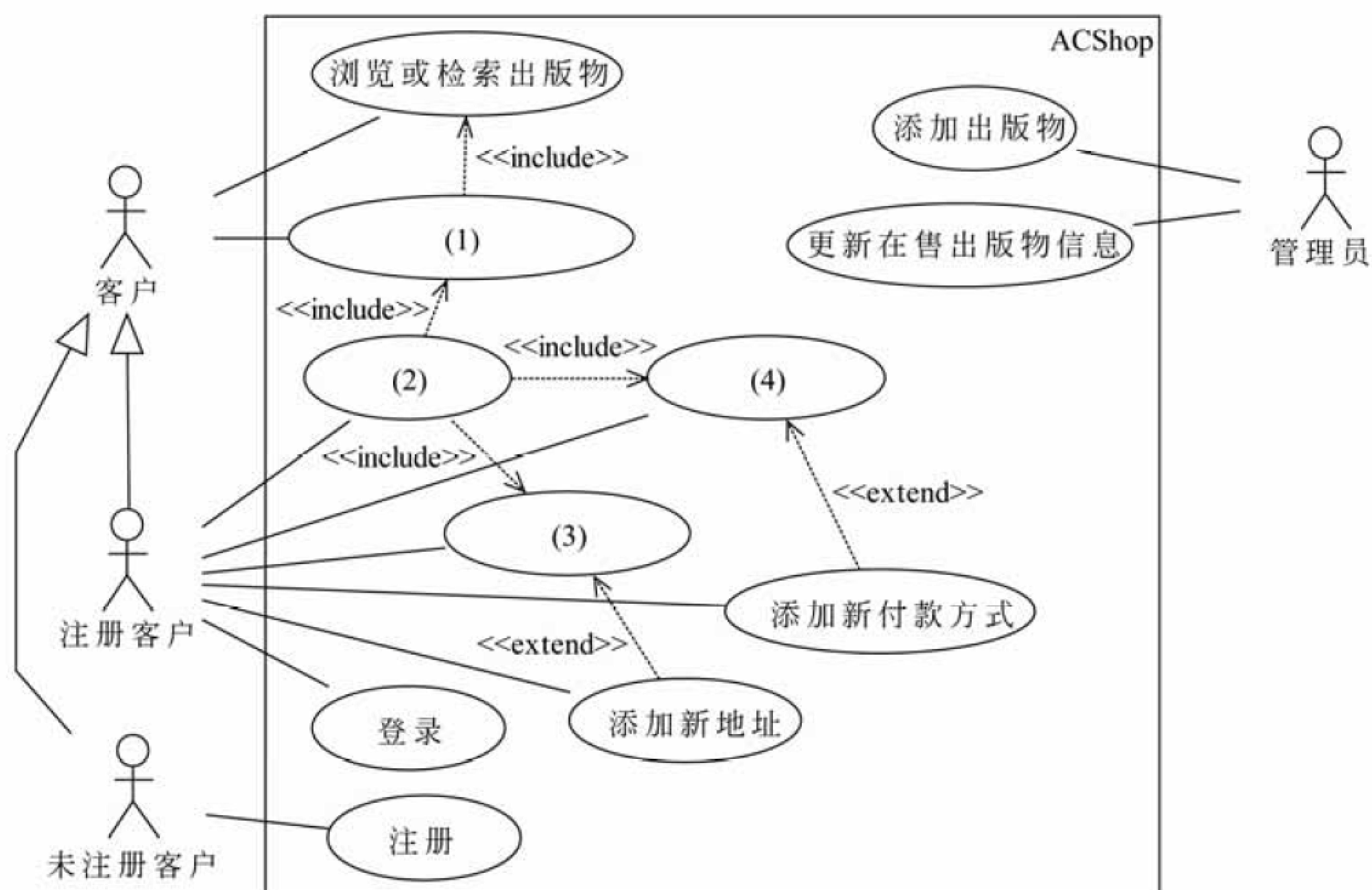


图 3-1　用例图

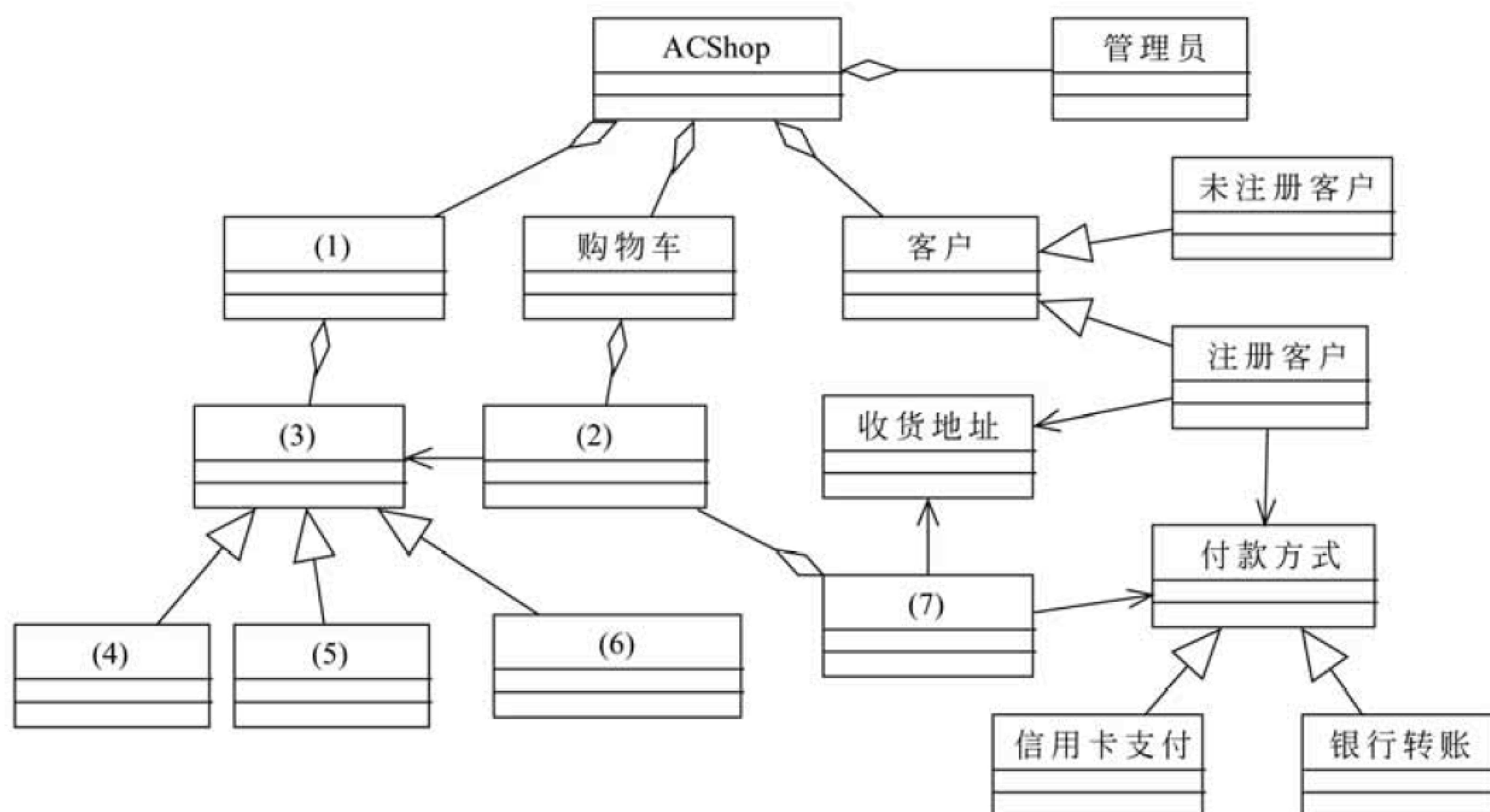


图 3-2　类图

试题三分析

本题属于经典的考题，主要考查面向对象分析方法与设计的基本概念。在建模方面，本题中涉及到了 UML 的用例图与类图。本题属于比较经典的考题，难度不大。

【问题 1】

本问题考查 UML 用例图，要求将图中缺失的用例（1）～（4）补充完整。解答此类题目的时候，根据给出的用例图对照说明中的功能需求描述，就可以完成。

首先（1）处的用例与参与者“客户”相关，而“客户”又分为“注册客户”和“未注册客户”，那么（1）处所代表的用例，是“注册客户”和“未注册客户”都具有的行为。由说明可知，（1）处的用例为“将（待购买）出版物添加到购物车”。

（2）～（3）处的用例与参与者“注册客户”相关，对照说明确定没有在用例图上表示出来的注册客户的行为即可，同时应注意用例（3）与“添加新地址”、用例（4）与“添加新付款方式”之间的扩展（extend）关系。根据说明可知，“注册客户”一个很重要的行为是“结账”，而这个行为在用例图恰好没有表示出来。再者，由说明中给出的结账操作的具体流程可知，结账操作中包含了选择地址和选择付款方式，与用例图中（2）和（3）、（2）和（4）之间的包含（include）关系对应，因此（2）处的用例为“结账”；而（3）处的用例为“选择收货地址”、（4）处的用例为“选择付款方式”。

【问题 2】

扩展是用例之间的一种关联关系。如果一个用例明显地混合了两种或两种以上的不同场景，即根据情况可能发生多种分支，则可以将这个用例分为一个基本用例和一个或多个扩展用例，这样使描述可能更加清晰。

用例（3）和（4）在结账操作的流程中给出了详细的描述：“如果没有地址信息，可以添加新地址信息”、“若没有付款方式信息，则可以添加新付款方式”。所以用例“添加新地址”和“添加新付款方式”分别是用例（3）和（4）的一种分支情况，其扩展点就是分支条件。所以“添加新地址”的扩展条件：地址信息为空或没有地址信息；“添加新付款方式”的扩展条件：付款方式信息为空或没有付款方式信息。

【问题 3】

本问题考查 UML 的类图，要求将图中缺失的类补充完整，是比较传统的考法。在解答此题时，可以先关注一下需要填写的类之间的关系。由类图可知，主要是两大类关系：聚集关系和继承关系。由说明可知，在题目中存在着 3 组继承关系：“ACShop 在线销售的学术出版物包括论文、学术报告或讲座资料等”；“ACShop 的客户分为两种：未注册客户和注册客户”；“ACShop 支持信用卡付款或银行转账两种方式”。后 2 组继承关系已经在类图中给出了，所以空（3）～（6）处要表达的就是第 1 组继承关系。由此可知，空（3）处应填入“（学术）出版物”，（4）～（6）处分别是“论文”“学术报告”和“讲座资料”。类（3）和类（1）之间是聚集关系，而现在已经知道类（3）表示的是“出版物”。由说明可知，与“出版物”之间具有聚集关系的应该是“出版物目录”，因此

（1）处应填入“出版物目录”。

类（2）与类“购物车”之间具有聚集关系，购物车中包含的是“待购买的出版物”，因此（2）处应填入“待购买的出版物”。由此也可以确定（7）处应该填入的类是“订单”。

试题三参考答案

【问题 1】

（1）将（待购买）出版物添加到购物车

（2）结账

（3）选择收货地址

（4）选择付款方式

【问题 2】

“添加新地址”的扩展条件：地址信息为空或没有地址信息。

“添加新付款方式”的扩展条件：付款方式信息为空或没有付款方式信息。

【问题 3】

（1）目录或出版物目录 （2）待购买的出版物 （3）出版物

（4）论文 （5）学术报告 （6）讲座资料

（7）订单

注：（4）～（6）答案次序可以互换。

试题四（共 15 分）

阅读下列说明和 C 代码，回答问题 1 至问题 3，将解答写在答题纸的对应栏内。

【说明】

计算两个字符串 x 和 y 的最长公共子串（Longest Common Substring）。

假设字符串 x 和字符串 y 的长度分别为 m 和 n，用数组 c 的元素 $c[i][j]$记录 x 中前 i 个字符和 y 中前 j 个字符的最长公共子串的长度。

$c[i][j]$满足最优子结构，其递归定义为：

$$c[i][j]=\begin{cases}c[i-1][j-1]+1 & \text{若 } i>0 \text{ 且 } j>0 \text{ 且 } x[i]=y[j]\\ 0 & \text{其他}\end{cases}$$

计算所有 $c[i][j]$（$0\leqslant i\leqslant m, 0\leqslant j\leqslant n$）的值，值最大的 $c[i][j]$即为字符串 x 和 y 的最长公共子串的长度。根据该长度即 i 和 j，确定一个最长公共子串。

【C 代码】

（1）常量和变量说明

x，y：长度分别为 m 和 n 的字符串。

$c[i][j]$：记录 x 中前 i 个字符和 y 中前 j 个字符的最长公共子串的长度。

max：x 和 y 的最长公共子串的长度。

maxi，maxj：分别表示 x 和 y 的某个最长公共子串的最后一个字符在 x 和 y 中

的位置（序号）。

（2）C 程序

```
#include <stdio.h>
#include <string.h>

int c[50][50];
int maxi;
int maxj;

int lcs(char *x, int m, char *y, int n)  {
    int i, j;
    int max = 0;
    maxi = 0;
    maxj = 0;

    for ( i = 0; i <= m; i++ )    c[i][0] = 0;
    for ( i = 1; i <= n; i++ )     c[0][i] = 0;
    for ( i = 1; i <= m; i++ )  {
      for ( j = 1; j <= n; j++ )   {
        if ( ____(1)____ )  {
          c[i][j] = c[i - 1][j - 1] + 1;
          if( max < c[i][j] ) {
            ____(2)____ ;
            maxi = i;
            maxj = j;
          }
        }
        else  ____(3)____ ;
      }
    }
    return max;
}
void printLCS(int max, char *x){
     int i = 0;
     if (max == 0)    return;
     for (____(4)____; i < maxi; i++)
        printf("%c", x[i]);
}
void main() {
char* x = "ABCADAB";
```

```
    char* y = "BDCABA";
    int max = 0;
    int m = strlen(x);
    int n = strlen(y);

    max = lcs(x, m, y, n);
    printLCS(max, x);
  }
```

【问题 1】（8 分）

根据以上说明和 C 代码，填充 C 代码中的空（1）～（4）。

【问题 2】（4 分）

根据题干说明和以上 C 代码，算法采用了＿（5）＿设计策略。

分析时间复杂度为＿（6）＿（用 O 符号表示）。

【问题 3】（3 分）

根据题干说明和以上 C 代码，输入字符串 x = "ABCADAB"，y = "BDCABA"，则输出为＿（7）＿。

试题四分析

本题考查算法设计与分析和 C 语言实现算法的相关技术。

此类题目要求考生认真阅读题目，首先理解问题以及求解问题的算法思路。

【问题 1】

根据题干说明，给出的问题具有最优子结构，考生应该能想到该题用动态规划或者贪心求解。一般在给出递归定义最优解时，已经比较清楚地给出要用动态规划方法，并且根据给出的 C 程序，可知以自底向上的方式进行计算，即先求小规模问题的，再求规模更大的问题的解。进入到 C 程序内部，函数 lcs 是计算 c 数组，并确定其最大的元素。在两重循环内，应该是递归公式的迭代求解过程，因此空(1)处填入“x[i - 1] == y[j - 1]”；若当前的最大长度小于 c[i][j]，则应该更新当前最大长度，即空(2)处填入“max = c[i][j]”；空(3)前面是 else 与 if 对应，即是 x[i - 1]≠y[j - 1]的情况，根据递归式此处填入“c[i][j] = 0”；函数 printLCS 是根据函数 lcs 计算的结果输出最长公共子串，长度为 max，在串 x 中的最后位置是 maxi，而在串 y 中的最后位置是 maxj，因此，空(4)填入“i = maxi – max”。

【问题 2】

根据【问题 1】中的分析，已知算法采用动态规划技术，算法的时间复杂度分析过程为：

（1）函数 lcs 中，有两个一重循环和一个两重循环，时间复杂度为 m+n+mn；

（2）函数 printLCS 中，有一个一重循环，时间复杂度为 m（或 n）。

故算法的时间复杂度为 O（mn）。

【问题 3】

根据题干和 C 代码，计算出下表的值。

表 1　二维数组 c

		0	1 B	2 D	3 C	4 A	5 B	6 A
	0	0	0	0	0	0	0	0
A	1	0	0	0	0	1	0	1
B	2	0	1	0	0	0	2	0
C	3	0	0	0	1	0	0	0
A	4	0	0	0	0	2	0	1
D	5	0	0	1	0	0	0	0
A	6	0	0	0	0	1	0	1
B	7	0	1	0	0	0	2	0

最大值为 2。在计算过程中，我们记录第一个最大值，即表中阴影部分元素，因此得到最长公共子串为 AB。

试题四参考答案

【问题 1】

（1）x[i - 1] == y[j - 1]

（2）max = c[i][j]

（3）c[i][j] = 0

（4）i = maxi - max

【问题 2】

（5）动态规划　　（6）O(m×n) 或 O(mn)

【问题 3】

（7）AB

试题五（15 分）

阅读下列说明和 C++代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

某大型购物中心欲开发一套收银软件，要求其能够支持购物中心在不同时期推出的各种促销活动，如打折、返利（例如，满 300 返 100）等。现采用策略（Strategy）模式实现该要求，得到如图 5-1 所示的类图。

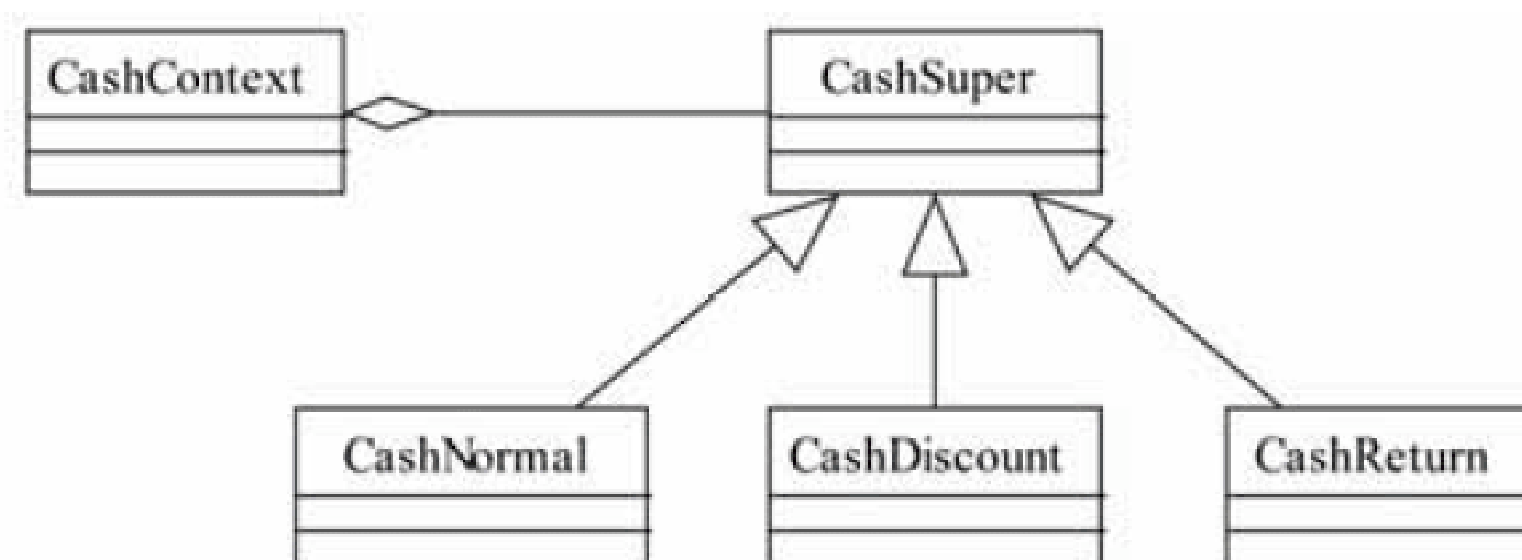


图 5-1　策略模式类图

【C++代码】

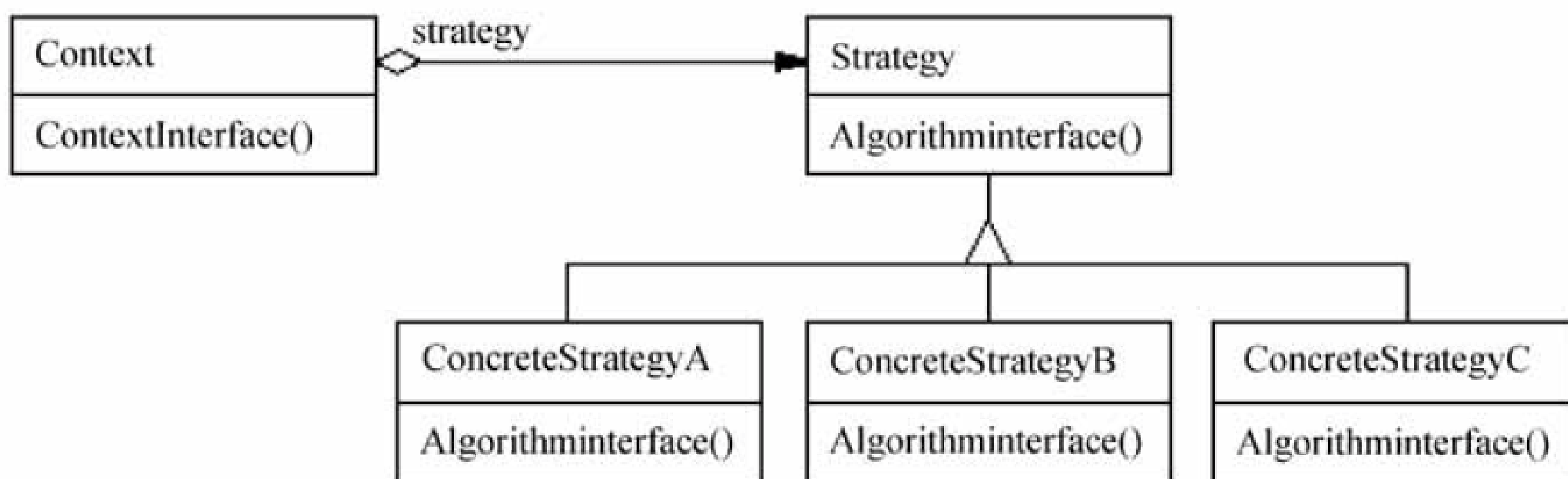
```
#include <iostream>
using namespace std;
enum TYPE { NORMAL, CASH_DISCOUNT, CASH_RETURN};
class CashSuper {
public:
   ____(1)____ ;
};
class CashNormal : public CashSuper {    //正常收费子类
public:
   double acceptCash(double money) {   return money;   }
};
class CashDiscount : public CashSuper {
private:
   double moneyDiscount;   //折扣率
public:
   CashDiscount(double discount) {     moneyDiscount = discount;  }
   double acceptCash(double money) {   return money * moneyDiscount;  }
};
class CashReturn : public CashSuper {    //满额返利
private:
   double moneyCondition;   //满额数额
   double moneyReturn;      //返利数额
public:
   CashReturn(double moneyCondition, double moneyReturn) {
     this->moneyCondition = moneyCondition;
     this->moneyReturn = moneyReturn;
   }
   double acceptCash(double money) {
     double result = money;
     if(money >= moneyCondition)
        result = money - (int)(money / moneyCondition ) * moneyReturn;
     return result;
  }
};
class CashContext {
private:
   CashSuper *cs;
public:
   CashContext(int type)  {
```

```
        switch(type) {
            case NORMAL:      //正常收费
                ____(2)____;
                break;
             case CASH_RETURN:     // 满 300 返 100
                ____(3)____;
                break;
             case CASH_DISCOUNT:    // 打 8 折
                ____(4)____;
                break;
        }
    }
    double GetResult(double money) {
        ____(5)____;
    }
};
// 此处略去 main()函数
```

试题五分析

本题考查策略（Strategy）模式的基本概念和应用。

Strategy 模式的设计意图是，定义一系列的算法，把它们一个个封装起来，并且使它们可以相互替换。此模式使得算法可以独立于使用它们的客户而变化，其结构图如下图所示。



- Strategy（策略）定义所有支持的算法的公共接口。Context 使用这个接口来调用某 ConcreteStrategy 定义的算法。
- ConcreteStrategy（具体策略）以 Strategy 接口实现某具体算法。
- Context（上下文）用一个 ConcreteStrategy 对象来配置；维护一个对 Strategy 对象的引用；可定义一个接口来让 Strategy 访问它的数据。

Strategy 模式适用于：

- 许多相关的类仅仅是行为有异。“策略”提供了一种用多个行为中的一个行为来配置一个类的方法。

- 需要使用一个算法的不同变体。例如，定义一些反应不同空间的空间/时间权衡的算法。当这些变体实现为一个算法的类层次时，可以使用策略模式。
- 算法使用客户不应该知道的数据。可使用策略模式以避免暴露复杂的、与算法相关的数据结构。

一个类定义了多种行为，并且这些行为在这个类的操作中以多个条件语句的形式出现，将相关的条件分支移入它们各自的 Strategy 类中，以代替这些条件语句。

本题中类 CashSuper 对应于上图中的类 Strategy，类 CashNormal、CashDiscout 和 CashReturn 分别代表 3 种不同的具体促销策略。CashSuper 类提供其 3 个子类的公共操作接口，由子类给出 3 种不同促销策略的具体实现。在 C++语言中，可以采用继承+（纯）虚拟函数来实现。从 3 个子类 CashNormal、CashDiscout 和 CashReturn 的代码可以看出，公共操作接口为 double acceptCash(double money)。由于不需要父类 CashSuper 提供任何的促销实现方式，所以这里采用纯虚拟函数。应填入空（1）处的语句是“virtual double acceptCash(double money) = 0”。

空（2）～（4）都出现在类 CashContext 中，该类对应于上图中的类 Context，其作用是依据策略对象来调用不同的策略算法。因此空（2）～（4）是根据不同的 case 分支来创建不同的策略对象。由此可知空（2）～（4）分别应填入“cs = new CashNormal()”“cs = new CashReturn(300, 100)”和“cs = new CashDiscount(0.8)”。

方法 GetResult 是对接口的调用，从而计算出采用不同促销策略之后应付的费用，这里需要通过 CashSuper 的对象 cs 来调用公共操作接口，因此第（5）空应填入“return cs->acceptCash(money)”。

试题五参考答案

（1）virtual double acceptCash(double money) = 0

（2）cs = new CashNormal()

（3）cs = new CashReturn(300, 100)

（4）cs = new CashDiscount(0.8)

（5）return cs->acceptCash(money)

试题六（15 分）

阅读下列说明和 Java 代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

某大型购物中心欲开发一套收银软件，要求其能够支持购物中心在不同时期推出的各种促销活动，如打折、返利（例如，满 300 返 100）等。现采用策略（Strategy）模式实现该要求，得到如图 6-1 所示的类图。

【Java 代码】

```
import java.util.*;
```

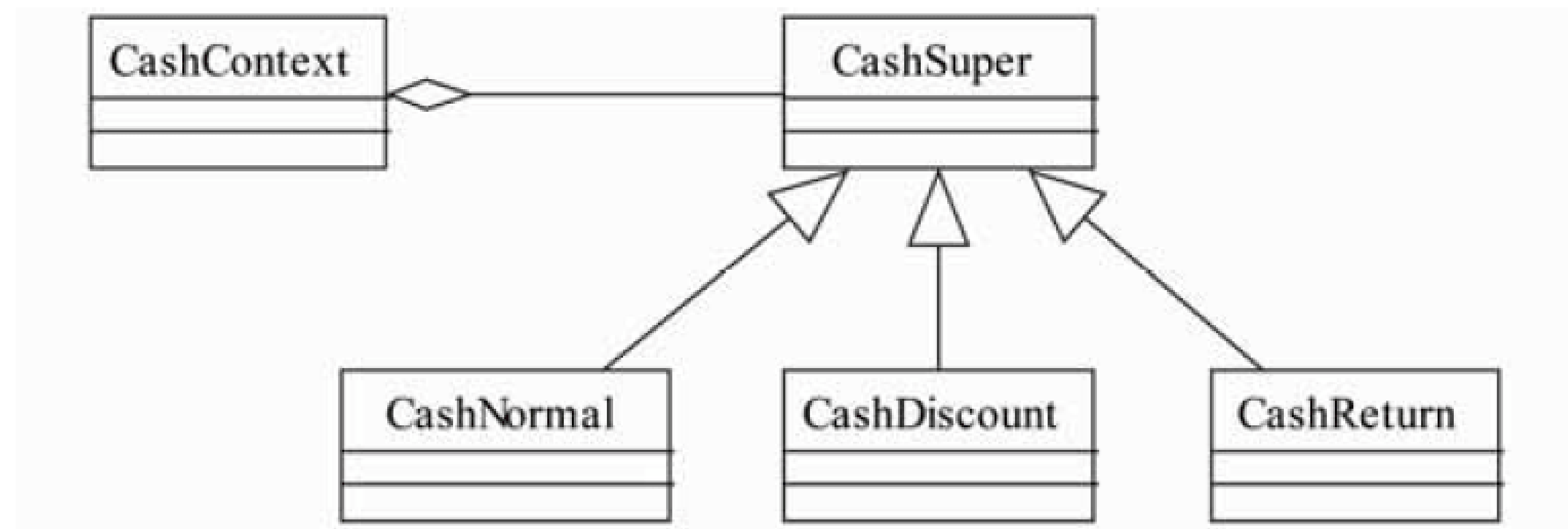


图 6-1　策略模式类图

```
enum TYPE { NORMAL, CASH_DISCOUNT, CASH_RETURN};
interface CashSuper {
  public ____(1)____ ;
}

class CashNormal implements CashSuper {      //正常收费子类
  public double acceptCash(double money) {
     return money;
  }
}

class CashDiscount implements CashSuper {
  private double moneyDiscount; // 折扣率
  public CashDiscount(double  moneyDiscount) {
     this.moneyDiscount = moneyDiscount;
  }
  public double acceptCash(double money) {
     return money * moneyDiscount;
  }
}
class CashReturn implements CashSuper {      //满额返利
  private double moneyCondition;
  private double moneyReturn;
  public CashReturn(double moneyCondition, double moneyReturn) {
     this.moneyCondition = moneyCondition;   //满额数额
     this.moneyReturn = moneyReturn;         //返利数额
  }
  public double acceptCash(double money) {
     double result = money;
     if(money >= moneyCondition)
        result = money - Math.floor(money / moneyCondition ) * moneyReturn;
```

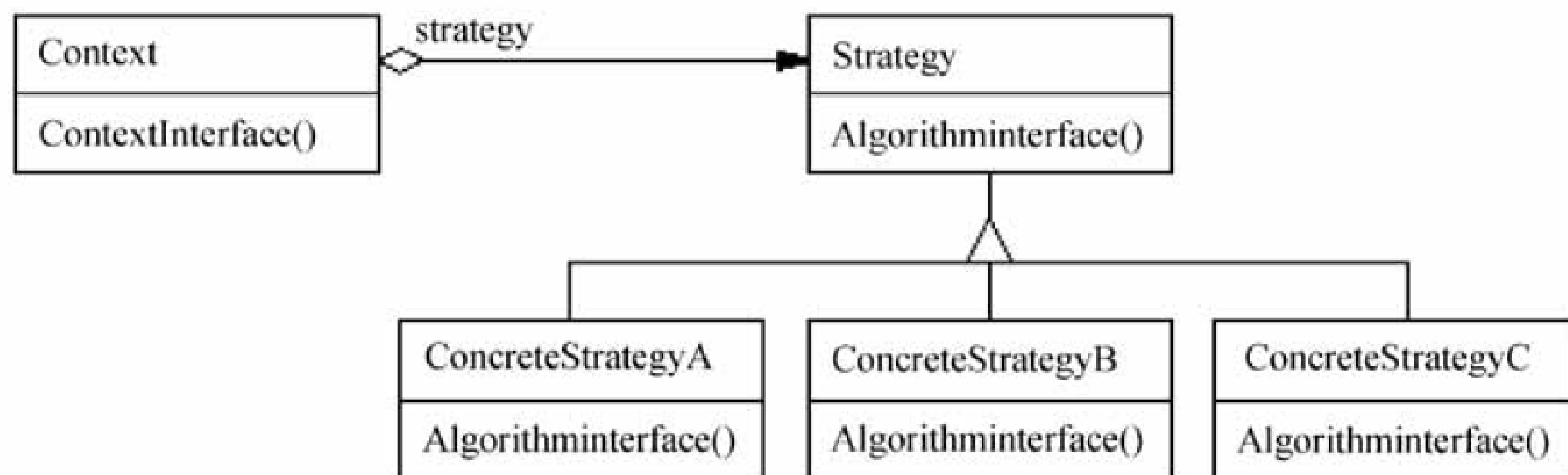
```
        return result;
      }
    }

    class CashContext {
      private CashSuper cs;
      private TYPE  t;
      public CashContext(TYPE t)  {
             switch(t) {
             case NORMAL:      //  "正常收费"
               ____(2)____ ;
                break;
             case CASH_DISCOUNT:    //  "打 8 折"
               ____(3)____ ;
                break;
             case CASH_RETURN:   //  "满 300 返 100"
               ____(4)____ ;
                break;
          }
      }
      public double GetResult(double money) {
         ____(5)____ ;
      }
      // 此处略去 main()函数
    }
```

试题六分析

本题考查策略（Strategy）模式的基本概念和应用。

Strategy 模式的设计意图是，定义一系列的算法，把它们一个个封装起来，并且使它们可以相互替换。此模式使得算法可以独立于使用它们的客户而变化，其结构图如下图所示。



- Strategy（策略）定义所有支持的算法的公共接口。Context 使用这个接口来调用

某 ConcreteStrategy 定义的算法。

- ConcreteStrategy（具体策略）以 Strategy 接口实现某具体算法。
- Context（上下文）用一个 ConcreteStrategy 对象来配置；维护一个对 Strategy 对象的引用；可定义一个接口来让 Strategy 访问它的数据。

Strategy 模式适用于：

- 许多相关的类仅仅是行为有异。“策略”提供了一种用多个行为中的一个行为来配置一个类的方法。
- 需要使用一个算法的不同变体。例如，定义一些反应不同空间的空间/时间权衡的算法。当这些变体实现为一个算法的类层次时，可以使用策略模式。
- 算法使用客户不应该知道的数据。可使用策略模式以避免暴露复杂的、与算法相关的数据结构。

一个类定义了多种行为，并且这些行为在这个类的操作中以多个条件语句的形式出现，将相关的条件分支移入它们各自的 Strategy 类中，以代替这些条件语句。

本题中类 CashSuper 对应于上图中的类 Strategy，类 CashNormal、CashDiscout 和 CashReturn 分别代表 3 种不同的具体促销策略。CashSuper 类提供其 3 个子类的公共操作接口，由子类给出 3 种不同促销策略的具体实现。这里采用了 Java 中的接口（Interface）来实现。从 3 个子类 CashNormal、CashDiscout 和 CashReturn 的代码可以看出，公共操作接口为 double acceptCash(double money)，因此应填入空（1）处的语句是“double acceptCash(double money)”。

空（2）～（4）都出现在类 CashContext 中，该类对应于上图中的类 Context，其作用是依据策略对象来调用不同的策略算法。因此空（2）～（4）是根据不同的 case 分支来创建不同的策略对象。由此可知空（2）～（4）分别应填入“cs = new CashNormal()”、“cs = new CashDiscount(0.8)”和“cs = new CashReturn(300, 100)”。

方法 GetResult 是对接口的调用，从而计算出采用不同促销策略之后应付的费用，这里需要通过 CashSuper 的对象 cs 来调用公共操作接口，因此第（5）空应填入“return cs.acceptCash(money)”。

试题六参考答案

（1）double acceptCash(double money)

（2）cs = new CashNormal()

（3）cs = new CashDiscount(0.8)

（4）cs = new CashReturn(300, 100)

（5）return cs.acceptCash(money)

第 29 章　2016 上半年软件设计师上午试题分析与解答

试题（1）、（2）

VLIW 是__（1）__的简称。

（1）A．复杂指令系统计算机　　B．超大规模集成电路
　　C．单指令流多数据流　　　D．超长指令字

试题（20）分析

本题计算机系统基础知识。

VLIW 是超长指令字的缩写。

参考答案

（1）D

试题（2）

主存与 Cache 的地址映射方式中，__（2）__方式可以实现主存任意一块装入 Cache 中任意位置，只有装满才需要替换。

（2）A．全相联　　B．直接映射　　C．组相联　　D．串并联

试题（2）分析

本题考查计算机系统基础知识。

全相联映射是指主存中任一块都可以映射到 Cache 中任一块的方式，也就是说，当主存中的一块需调入 Cache 时，可根据当时 Cache 的块占用或分配情况，选择一个块给主存块存储，所选的 Cache 块可以是 Cache 中的任意一块。

直接相联映射方式是指主存的某块 j 只能映射到满足特定关系的 Cache 块 i 中。

全相联映射和直接相联映射方式的优缺点正好相反，也就是说，对于全相联映射方式来说为优点的恰是直接相联映射方式的缺点，而对于全相联映射方式来说为缺点的恰是直接相联映射方式的优点。

组相连映像了兼顾这两种方式的优点：主存和 Cache 按同样大小划分成块；主存和 Cache 按同样大小划分成组；主存容量是缓存容量的整数倍，将主存空间按缓冲区的大小分成区，主存中每一区的组数与缓存的组数相同；当主存的数据调入缓存时，主存与缓存的组号应相等，也就是各区中的某一块只能存入缓存的同组号的空间内，但组内各块地址之间则可以任意存放，即从主存的组到 Cache 的组之间采用直接映象方式；在两个对应的组内部采用全相联映象方式。

参考答案

（2）A

试题（3）

如果“2X”的补码是“90H”，那么 X 的真值是__（3）__。

（3）A．72　　B．–56　　C．56　　D．111

试题（3）分析

本题考查计算机系统基础知识。

先由补码“90H”得出其对应的真值，为负数，绝对值的二进制形式为 01110000，转换为十进制后等于–112，即 2x=–112，因此 x=–56。

参考答案

（3）B

试题（4）

移位指令中的__（4）__指令的操作结果相当于对操作数进行乘 2 操作。

（4）A．算术左移　　B．逻辑右移　　C．算术右移　　D．带进位循环左移

试题（4）分析

本题考查计算机系统基础知识。

算术移位时，对于负数，其符号位可能需要特殊处理，逻辑移位中没有符号的概念，只是二进制位序列。

算术左移等同于乘以 2 的操作。

参考答案

（4）A

试题（5）

内存按字节编址，从 A1000H 到 B13FFH 的区域的存储容量为__（5）__KB。

（5）A．32　　B．34　　C．65　　D．67

试题（5）分析

本题考查计算机系统基础知识。

结束地址和起始地址的差值再加 1 为存储单元的个数，即

B13FFH–A1000H+1=10400H

转换为十进制后等于 65536+1024=64KB+1KB=65K

参考答案

（5）C

试题（6）

以下关于总线的叙述中，不正确的是__（6）__。

（6）A．并行总线适合近距离高速数据传输

B．串行总线适合长距离数据传输

C．单总线结构在一个总线上适应不同种类的设备，设计简单且性能很高

D．专用总线在设计上可以与连接设备实现最佳匹配

试题（6）分析

本题考查计算机系统基础知识。

串行总线将数据一位一位传输，数据线只需要一根（如果支持双向需要 2 根），并行总线是将数据的多位同时传输（4 位，8 位，甚至 64 位，128 位），显然，并行总线的传输速度快，在长距离情况下成本高，串行传输的速度慢，但是远距离传输时成本低。

单总线结构在一个总线上适应不同种类的设备，通用性强，但是无法达到高的性能要求，而专用总线则可以与连接设备实现最佳匹配。

参考答案

（6）C

试题（7）

以下关于网络层次与主要设备对应关系的叙述中，配对正确的是＿（7）＿。

（7）A．网络层——集线器　　B．数据链路层——网桥
C．传输层——路由器　　D．会话层——防火墙

试题（7）分析

网络层的联网设备是路由器，数据链路层的联网设备是网桥和交换机，传输层和会话层主要是软件功能，都不需要专用的联网设备。

参考答案

（7）B

试题（8）

传输经过 SSL 加密的网页所采用的协议是＿（8）＿。

（8）A．HTTP　　B．HTTPS　　C．S-HTTP　　D．HTTP-S

试题（8）分析

本题考查 HTTPS 基础知识。

HTTPS（Hyper Text Transfer Protocol over Secure Socket Layer），是以安全为目标的 HTTP 通道，即使用 SSL 加密算法的 HTTP。

参考答案

（8）B

试题（9）

为了攻击远程主机，通常利用＿（9）＿技术检测远程主机状态。

（9）A．病毒查杀　　B．端口扫描　　C．QQ 聊天　　D．身份认证

试题（9）分析

本题考查网络安全中漏洞扫描基础知识。

通常利用通过端口漏洞扫描来检测远程主机状态，获取权限从而攻击远程主机。

参考答案

（9）B

试题（10）

某软件公司参与开发管理系统软件的程序员张某，辞职到另一公司任职，于是该项目负责人将该管理系统软件上开发者的署名更改为李某（接张某工作）。该项目负责人的行为＿（10）＿。

（10）A．侵犯了张某开发者身份权（署名权）

B．不构成侵权，因为程序员张某不是软件著作权人

C．只是行使管理者的权利，不构成侵权

D．不构成侵权，因为程序员张某现已不是项目组成员

试题（10）分析

《计算机软件保护条例》规定软件著作权人享有的权利，包括发表权、署名权、修改权、复制权、发行权、出租权、信息网络传播权、翻译权。署名权是指软件开发者为表明身份在自己开发的软件原件及其复制件上标记姓名的权利。法律法规规定署名权的根本目的，在于保障不同软件来自不同开发者这一事实不被人混淆，署名即是标记，旨在区别，区别的目的是为了有效保护软件著作权人的合法权益。署名彰显了开发者与软件之间存在关系的客观事实。因此，行使署名权应当奉行诚实的原则，应当符合有效法律行为的要件，否则会导致署名无效的后果。

署名权只能是真正的开发者和被视同开发者的法人和非法人团体才有资格享有，其他任何个人、单位和组织不得行使此项权利。所以，署名权还隐含着另一种权利，即开发者资格权。法律保护署名权，意味着法律禁止任何未参加开发人在他人开发的软件上署名。《计算机软件保护条例》规定“在他人开发的软件上署名或者更改他人开发的软件上的署名”的行为是侵权行为，这种行为侵犯了开发者身份权即署名权。

参考答案

（10）A

试题（11）

美国某公司与中国某企业谈技术合作，合同约定使用 1 项美国专利（获得批准并在有效期内），该项技术未在中国和其他国家申请专利。依照该专利生产的产品＿（11）＿需要向美国公司支付这件美国专利的许可使用费。

（11）A．在中国销售，中国企业

B．如果返销美国，中国企业不

C．在其他国家销售，中国企业

D．在中国销售，中国企业不

试题（11）分析

依照该专利生产的产品在中国或其他国家销售，中国企业不需要向美国公司支付这件美国专利的许可使用费。这是因为，该美国公司未在中国及其他国家申请该专利，不受中国及其他国家专利法的保护，因此，依照该专利生产的产品在中国及其他国家销售，

中国企业不需要向美国公司支付这件美国专利的许可使用费。

如果返销美国，需要向美国公司支付这件美国专利的许可使用费。这是因为，这件专利已在美国获得批准，因而受到美国专利法的保护，中国企业依照该专利生产的产品要在美国销售，则需要向美国公司支付这件美国专利的许可使用费。

参考答案

（11）D

试题（12）

以下媒体文件格式中，__（12）__是视频文件格式。

（12）A. WAV　　B. BMP　　C. MP3　　D. MOV

试题（12）分析

WAV 为微软公司开发的一种声音文件格式，它符合 RIFF(Resource Interchange File Format)文件规范。

BMP（Bitmap）是 Windows 操作系统中的标准图像文件格式，可以分成两类：设备相关位图（DDB）和设备无关位图（DIB）。它采用位映射存储格式，除了图像深度可选以外，不采用其他任何压缩。

MP3（Moving Picture Experts Group Audio Layer III）是一种音频压缩技术，它被设计用来大幅度地降低音频数据量。作为文件扩展名时表示该文件时一种音频格式文件。

MOV 即 QuickTime 影片格式，它是 Apple 公司开发的一种音频、视频文件格式，用于存储常用数字媒体类型。

参考答案

（11）D

试题（13）

以下软件产品中，属于图像编辑处理工具的软件是__（13）__。

（13）A. Powerpoint　　B. Photoshop　　C. Premiere　　D. Acrobat

试题（13）分析

Powerpoint 是微软公司的演示文稿软件。

Premiere 是一款常用的视频编辑软件，由Adobe公司推出。

Acrobat 是由Adobe公司开发的一款PDF(Portable Document Format)编辑软件。

Photoshop(简称 PS)，是由Adobe Systems 开发和发行的图像处理软件。

参考答案

（13）B

试题（14）

使用 150DPI 的扫描分辨率扫描一幅 3×4 英寸的彩色照片，得到原始的 24 位真彩色图像的数据量是__（14）__Byte。

（14）A．1800　　B．90000　　C．270000　　D．810000

试题（14）分析

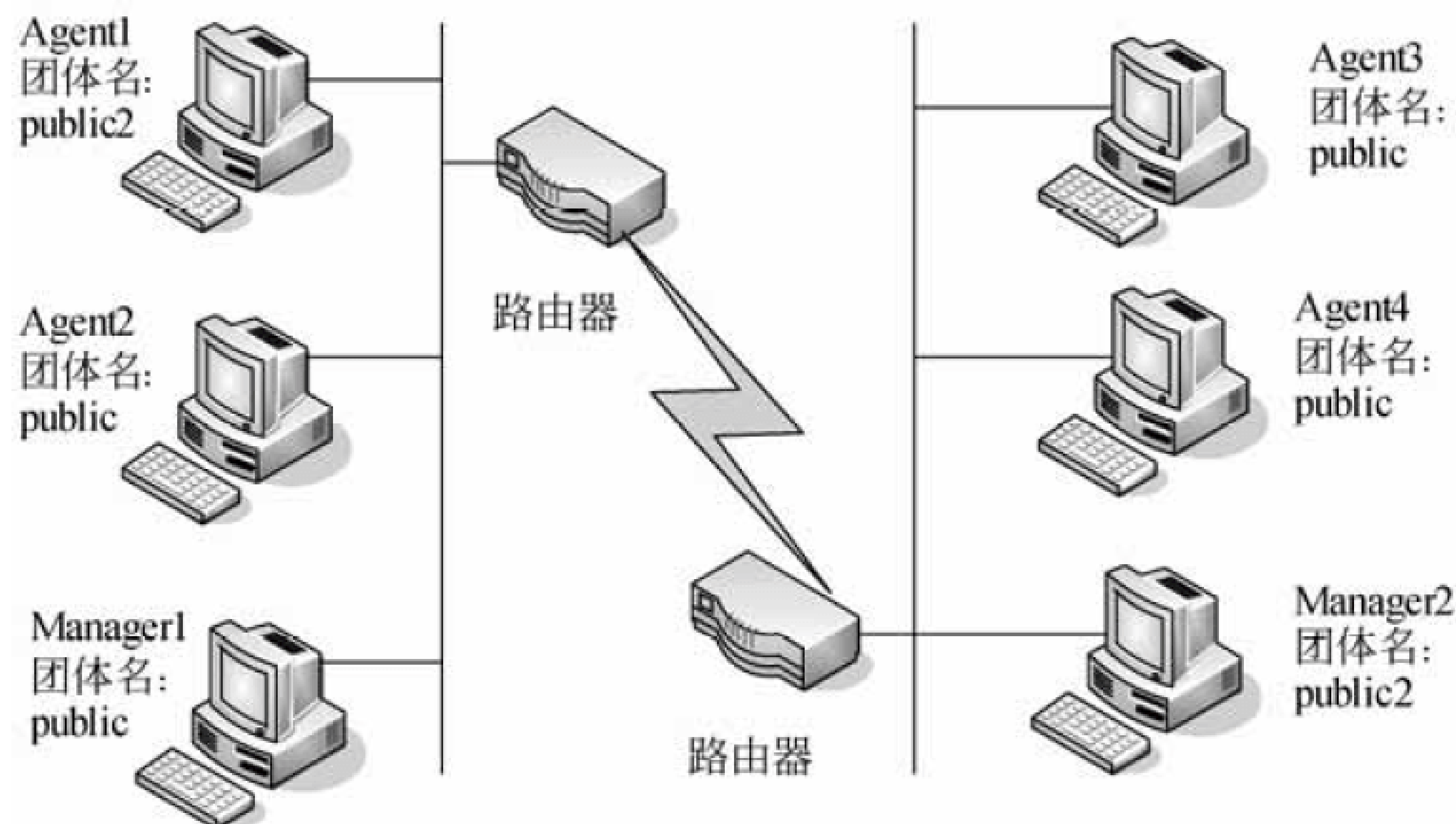
DPI（Dots Per Inch，每英寸点数）通常用来描述数字图像输入设备（如图像扫描仪）或点阵图像输出设备（点阵打印机）输入或输出点阵图像的分辨率。一幅 3×4 英寸的彩色照片在 150DPI 的分辨率下扫描得到原始的 24 位真彩色图像的数据量是（150×3）×（150×4）×24/8= 810000 字节。

参考答案

（14）D

试题（15）、（16）

某软件项目的活动图如下图所示，其中顶点表示项目里程碑，连接顶点的边表示包含的活动，边上的数字表示活动的持续时间（天），则完成该项目的最少时间为__（15）__天。活动 BD 最多可以晚开始__（16）__天而不会影响整个项目的进度。



（15）A．15　　B．21　　C．22　　D．24

（16）A．0　　B．2　　C．3　　D．5

试题（15）、（16）分析

本题考查软件项目管理的基础知识。

活动图是描述一个项目中各个工作任务相互依赖关系的一种模型，项目的很多重要特性可以通过分析活动图得到，如估算项目完成时间，计算关键路径和关键活动等。

根据上图计算出关键路径为 A-B-D-G-I- K-L，其长度为 22，关键路径上的活动均为关键活动。活动 BD 在关键路径上，因此松弛时间为 0。

参考答案

（15）C　（16）A

试题（17）、（18）

在结构化分析中，用数据流图描述__（17）__。当采用数据流图对一个图书馆管理系

统进行分析时，__（18）__是一个外部实体。

（17）A. 数据对象之间的关系，用于对数据建模

B. 数据在系统中如何被传送或变换，以及如何对数据流进行变换的功能或子功能，用于对功能建模

C. 系统对外部事件如何响应，如何动作，用于对行为建模

D. 数据流图中的各个组成部分

（18）A. 读者　　B. 图书　　C. 借书证　　D. 借阅

试题（17）、（18）分析

本题考查结构化分析的基础知识。

数据流图是结构化分析的一个重要模型，描述数据在系统中如何被传送或变换，以及描述如何对数据流进行变换的功能，用于功能建模。

数据流图中有四个要素：外部实体，也称为数据源或数据汇点，表示要处理的数据的输入来源或处理结果要送往何处，不属于目标系统的一部分，通常为组织、部门、人、相关的软件系统或者硬件设备；数据流表示数据沿箭头方向的流动；加工是对数据对象的处理或变换；数据存储在数据流中起到保存数据的作用，可以是数据库文件或者任何形式的数据组织。

根据上述定义和题干说明，读者是外部实体，图书和借书证是数据流，借阅是加工。

参考答案

（17）B　（18）A

试题（19）

软件开发过程中，需求分析阶段的输出不包括__（19）__。

（19）A. 数据流图　　B. 实体联系图

C. 数据字典　　D. 软件体系结构图

试题（19）分析

本题考查软件开发过程的基础知识。

结构化分析模型包括数据流图、实体联系图、状态迁移图和数据字典，因此这些模型是需求分析阶段的输出。而确定软件体系结构是在软件设计阶段进行的。

参考答案

（19）D

试题（20）

以下关于高级程序设计语言实现的编译和解释方式的叙述中，正确的是__（20）__。

（20）A. 编译程序不参与用户程序的运行控制，而解释程序则参与

B. 编译程序可以用高级语言编写，而解释程序只能用汇编语言编写

C. 编译方式处理源程序时不进行优化，而解释方式则进行优化

D. 编译方式不生成源程序的目标程序，而解释方式则生成

试题（20）分析

本题考查程序语言基础知识。

解释程序也称为解释器，它或者直接解释执行源程序，或者将源程序翻译成某种中间代码后再加以执行；而编译程序（编译器）则是将源程序翻译成目标语言程序，然后在计算机上运行目标程序。这两种语言处理程序的根本区别是：在编译方式下，机器上运行的是与源程序等价的目标程序，源程序和编译程序都不再参与目标程序的执行过程；而在解释方式下，解释程序和源程序（或其某种等价表示）要参与到程序的运行过程中，运行程序的控制权在解释程序。简单来说，在解释方式下，翻译源程序时不生成独立的目标程序，而编译器则将源程序翻译成独立保存的目标程序。

参考答案

（20）A

试题（21）

以下关于脚本语言的叙述中，正确的是<u>（21）</u>。

（21）A．脚本语言是通用的程序设计语言

B．脚本语言更适合应用在系统级程序开发中

C．脚本语言主要采用解释方式实现

D．脚本语言中不能定义函数和调用函数

试题（21）分析

本题考查程序语言基础知识。

维基百科上将脚本语言定义为“为了缩短传统的编写—编译—链接—运行过程而创建的计算机编程语言。通常具有简单、易学、易用的特色，目的是希望开发者以简单的方式快速完成某些复杂程序的编写工作。”

脚本语言一般运行在解释器或虚拟机中，便于移植，开发效率较高。

参考答案

（21）C

试题（22）

将高级语言源程序先转化为一种中间代码是现代编译器的常见处理方式。常用的中间代码有后缀式、<u>（22）</u>、树等。

（22）A．前缀码　　B．三地址码　　C．符号表　　D．补码和移码

试题（22）分析

本题考查程序语言基础知识。

“中间代码”是一种简单且含义明确的记号系统，可以有若干种形式，它们的共同特征是与具体的机器无关。最常用的一种中间代码是与汇编语言的指令非常相似的三地址码，其实现方式常采用四元式，另外还有后缀式、树等形式的中间代码。

参考答案

（22）B

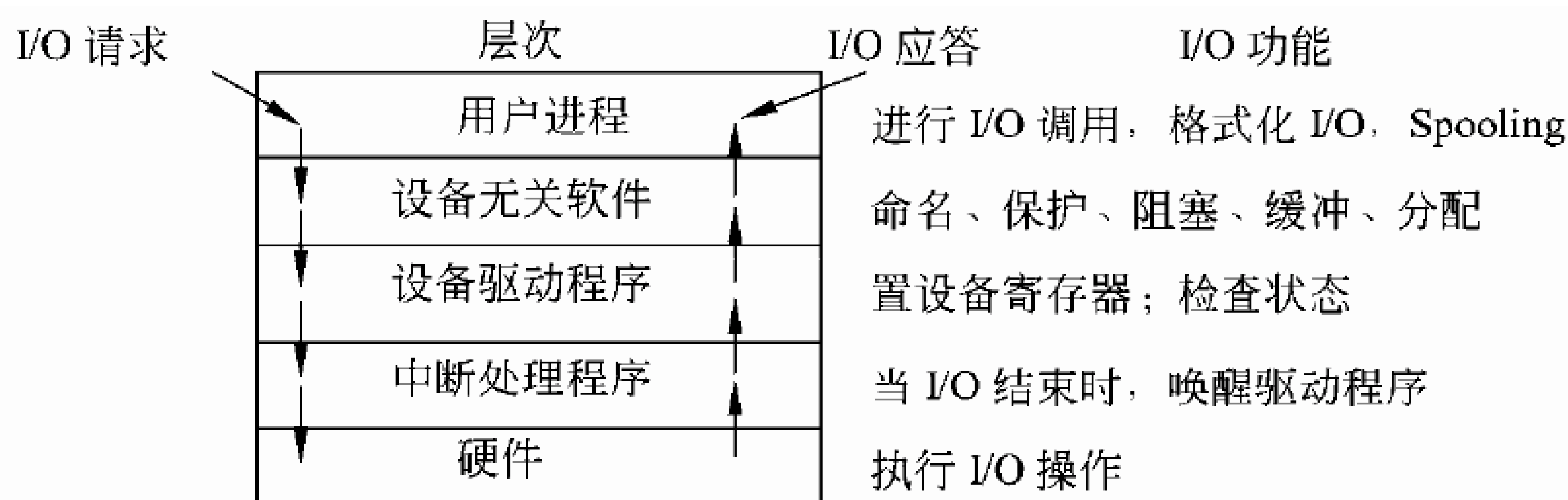
试题（23）

当用户通过键盘或鼠标进入某应用系统时，通常最先获得键盘或鼠标输入信息的是__（23）__程序。

（23）A．命令解释　　B．中断处理　　C．用户登录　　D．系统调用

试题（23）分析

I/O 设备管理软件一般分为 4 层：中断处理程序、设备驱动程序、与设备无关的系统软件和用户级软件。至于一些具体分层时细节上的处理，是依赖于系统的，没有严格的划分，只要有利于设备独立这一目标，可以为了提高效率而设计不同的层次结构。I/O 软件的所有层次及每一层的主要功能如下图所示。



图中的箭头给出了 I/O 部分的控制流。当用户通过键盘或鼠标进入某应用系统时，通常最先获得键盘或鼠标输入信息的程序是中断处理程序。

参考答案

（23）B

试题（24）

在 Windows 操作系统中，当用户双击“IMG_20160122_103.jpg”文件名时，系统会自动通过建立的__（24）__来决定使用什么程序打开该图像文件。

（24）A．文件　　B．文件关联　　C．文件目录　　D．临时文件

试题（24）分析

本题考查 Windows 操作系统文件管理方面的基础知识。

当用户双击一个文件名时，Windows 系统通过建立的文件关联来决定使用什么程序打开该文件。例如系统建立了“Windows 照片查看器”或“11view”程序打开扩展名为“.jpg”类型的文件关联，那么当用户双击“IMG_20160122_103.jpg”文件时，Windows 先执行“Windows 照片查看器”或“11view”程序，然后打开“IMG_20160122_103.jpg”文件。

参考答案

（24）B

试题（25）

某磁盘有 100 个磁道，磁头从一个磁道移至另一个磁道需要 6ms。文件在磁盘上非

连续存放，逻辑上相邻数据块的平均距离为 10 个磁道，每块的旋转延迟时间及传输时间分别为 100ms 和 20ms，则读取一个 100 块的文件需要__（25）__ms。

（25）A．12060　　B．12600　　C．18000　　D．186000

试题（25）分析

本题考查操作系统中设备管理的基本知识。

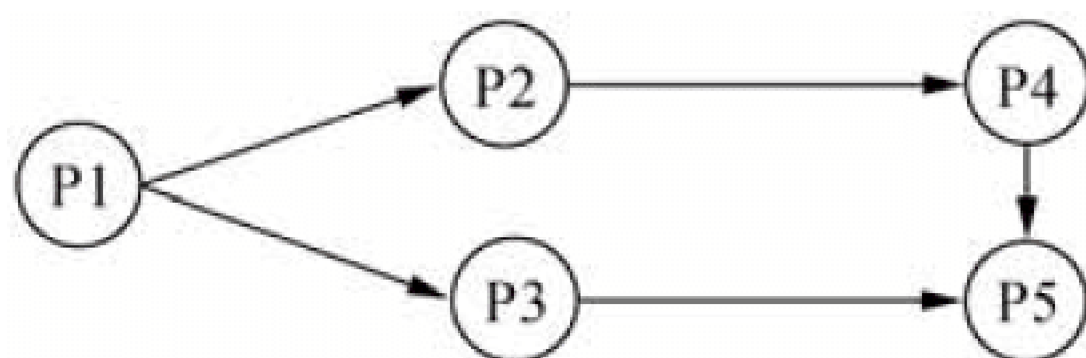
访问一个数据块的时间应为寻道时间加旋转延迟时间及传输时间之和。根据题意，每块的旋转延迟时间及传输时间共需 120ms，磁头从一个磁道移至另一个磁道需要 6ms，但逻辑上相邻数据块的平均距离为 10 个磁道，即读完一个数据块到下一个数据块寻道时间需要 60ms。通过上述分析，本题访问一个数据块的时间 T=120ms+60ms =180ms，而读取一个 100 块的文件共需要 18000ms。

参考答案

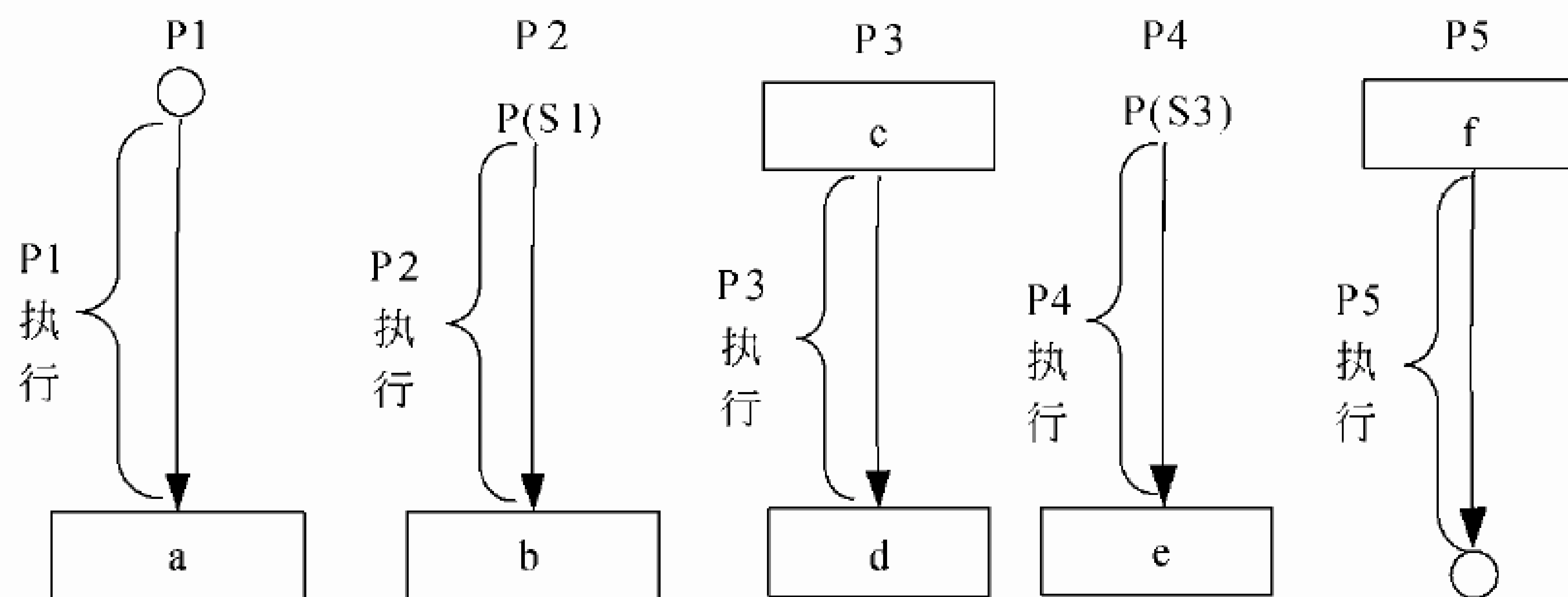
（25）C

试题（26）～（28）

进程 P1、P2、P3、P4 和 P5 的前趋图如下图所示：



若用 PV 操作控制进程 P1、P2、P3、P4 和 P5 并发执行的过程，则需要设置 5 个信号量 S1、S2、S3、S4 和 S5，且信号量 S1～S5 的初值都等于零。下图中 a 和 b 处应分别填入__（26）__；c 和 d 处应分别填入__（27）__；e 和 f 处应分别填入__（28）__。



（26）A．V（S1）P（S2）和 V（S3）　　B．P（S1）V（S2）和 V（S3）
　　C．V（S1）V（S2）和 V（S3）　　D．P（S1）P（S2）和 V（S3）

（27）A．P（S2）和 P（S4 ）　　B．P（S2）和 V（S4）
　　C．V（S2）和 P（S4）　　D．V（S2）和 V（S4）

（28）A．P（S4）和 V（S4）V（S5）　　B．V（S5）和 P（S4）P（S5）
　　C．V（S3）和 P（S4）P（S5）　　D．P（S3）和 P（S4）P（S5）

试题（26）～（28）分析

根据前驱图，P1 进程执行完需要通知 P2 和 P3 进程，故需要利用 V（S1）V（S2）操作通知 P2 和 P3 进程，所以空 a 应填 V（S1）V（S2）；P2 进程执行完需要通知 P4 进程，所以空 b 应填 V（S3）。

根据前驱图，P3 进程运行前需要等待 P1 进程的结果，故执行程序前要先利用 1 个 P 操作，而 P3 进程运行结束需要通知 P5 进程。根据排除法可选项只有选项 B 和选项 C。又因为 P3 进程运行结束后需要利用 1 个 V 操作通知 P5 进程，根据排除法可选项只有选项 B 满足要求。

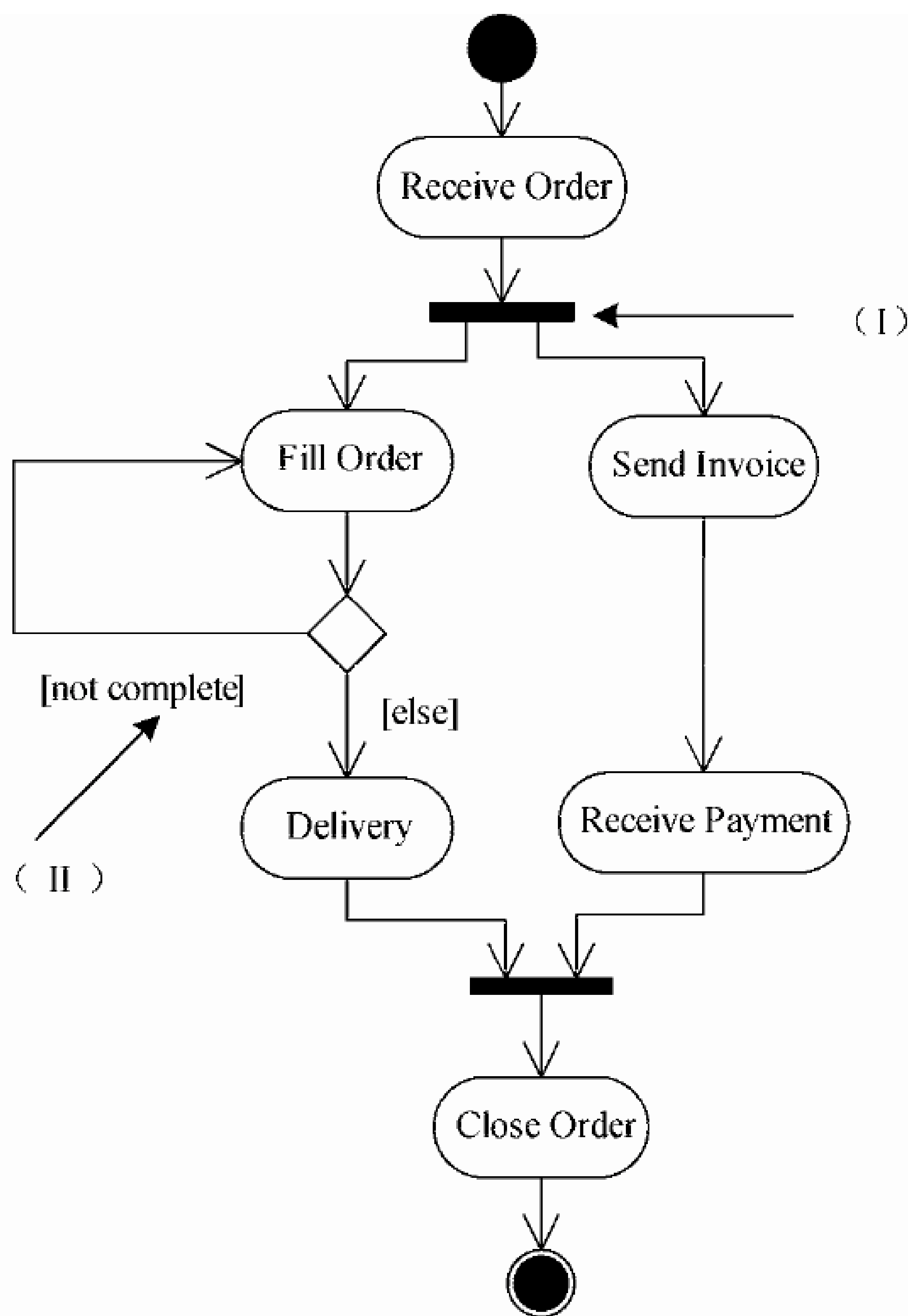
根据前驱图，P4 进程执行结束需要利用 1 个 V 操作通知 P5 进程，故空 e 处需要 1 个 V 操作；P5 进程执行前需要等待 P3 和 P4 进程的结果，故空 f 处需要 2 个 P 操作。根据排除法可选项只有选项 B 和选项 C 能满足要求。根据试题（27）分析可知，P3 进程运行结束是利用 V（S4）通知 P5 进程，故 P4 进程运行结束是利用 V（S5）通知 P5 进程。

参考答案

（26）C （27）B （28）B

试题（29）

如下图所示，模块 A 和模块 B 都访问相同的全局变量和数据结构，则这两个模块之间的耦合类型为 （29） 耦合。



（29）A．公共　　B．控制　　C．标记　　D．数据

试题（29）分析

本题考查软件设计基础知识。

模块独立性是创建良好设计的一个重要原则，一般采用模块间的耦合和模块的内聚两个准则来进行度量。耦合程度越低，内聚程度越高，则模块的独立性越好。存在多种模块之间的耦合类型，从低到高依次为非直接耦合、数据耦合、标记耦合、控制耦合、外部耦合、公共耦合和内容耦合。 其中，

公共耦合是指一组模块都访问同一公共数据环境；

控制耦合是指一个模块通过传送开关、标志、名字等控制信息，明显地控制选择另一个模块的功能；

标记耦合是一组模块通过参数表传递记录信息；

数据耦合是一个模块访问另一个模块时，彼此之间通过数据参数（不是控制参数，公共数据结构或外部变量），来交换输入输出信息。

参考答案

（29）A

试题（30）

以下关于增量开发模型的叙述中，不正确的是＿（30）＿。

（30）A．不必等到整个系统开发完成就可以使用

B．可以使用较早的增量构件作为原型，从而获得稍后的增量构件需求

C．优先级最高的服务先交付，这样最重要的服务接受最多的测试

D．有利于进行好的模块划分

试题（30）分析

本题考查开发过程模型的基础知识。

增量开发模型将软件产品分解成一系列的增量构件，在增量开发中逐步加入。其优点主要有：能在较短的时间内交付可以使用的部分产品；逐步增加的产品功能可以使用户有充裕的时间学习和适应新产品；优先级最高的功能首先交付，这意味着最重要的功能经过最多的测试。但是要求对要开发的系统进行精心的分析和设计。

参考答案

（30）D

试题（31）

在设计软件的模块结构时，＿（31）＿不能改进设计质量。

（31）A．模块的作用范围应在其控制范围之内

B．模块的大小适中

C．避免或减少使用病态连接（从中部进入或访问一个模块）

D．模块的功能越单纯越好

试题（31）分析

本题考查软件设计的基础知识。

在设计软件的模块结构时，有一些启发式原则可以改进设计。如完善模块功能、消除重复功能、模块的作用范围应在其控制范围之内、尽可能减少高扇出结构，随着深度增大扇入、避免或减少使用病态连接等等。模块规模大小应适中。模块单一的功能可以提高其内聚性，但同时考虑与其他模块的耦合程度，因此不是模块功能越单纯越好。

参考答案

（31）D

试题（32）、（33）

软件体系结构的各种风格中，仓库风格包含一个数据仓库和若干个其他构件。数据仓库位于该体系结构的中心，其他构件访问该数据仓库并对其中的数据进行增、删、改等操作。以下关于该风格的叙述中，不正确的是__（32）__。__（33）__不属于仓库风格。

（32）A. 支持可更改性和可维护性　　B. 具有可复用的知识源

C. 支持容错性和健壮性　　D. 测试简单

（33）A. 数据库系统　B. 超文本系统　C. 黑板系统　D. 编译器

试题（32）、（33）分析

本题考查软件体系结构的基础知识。

仓库风格是一种软件体系结构，其中包含一个数据仓库和若干个其他构件。数据仓库位于该体系结构的中心，其他构件访问该数据仓库并对其中的数据进行增、删、改等操作。数据库系统、超文本系统和黑板系统都属于仓库风格。

该体系结构的优点包括：

① 对可更改性和可维护性的支持；

② 可复用的知识源；

③ 支持容错性和健壮性。

缺点包括：

① 测试困难；

② 不能保证有好的解决方案；

③ 难以建立好的控制策略；

④ 低效；

⑤ 昂贵的开发工作；

⑥ 缺少对并行机制的支持。

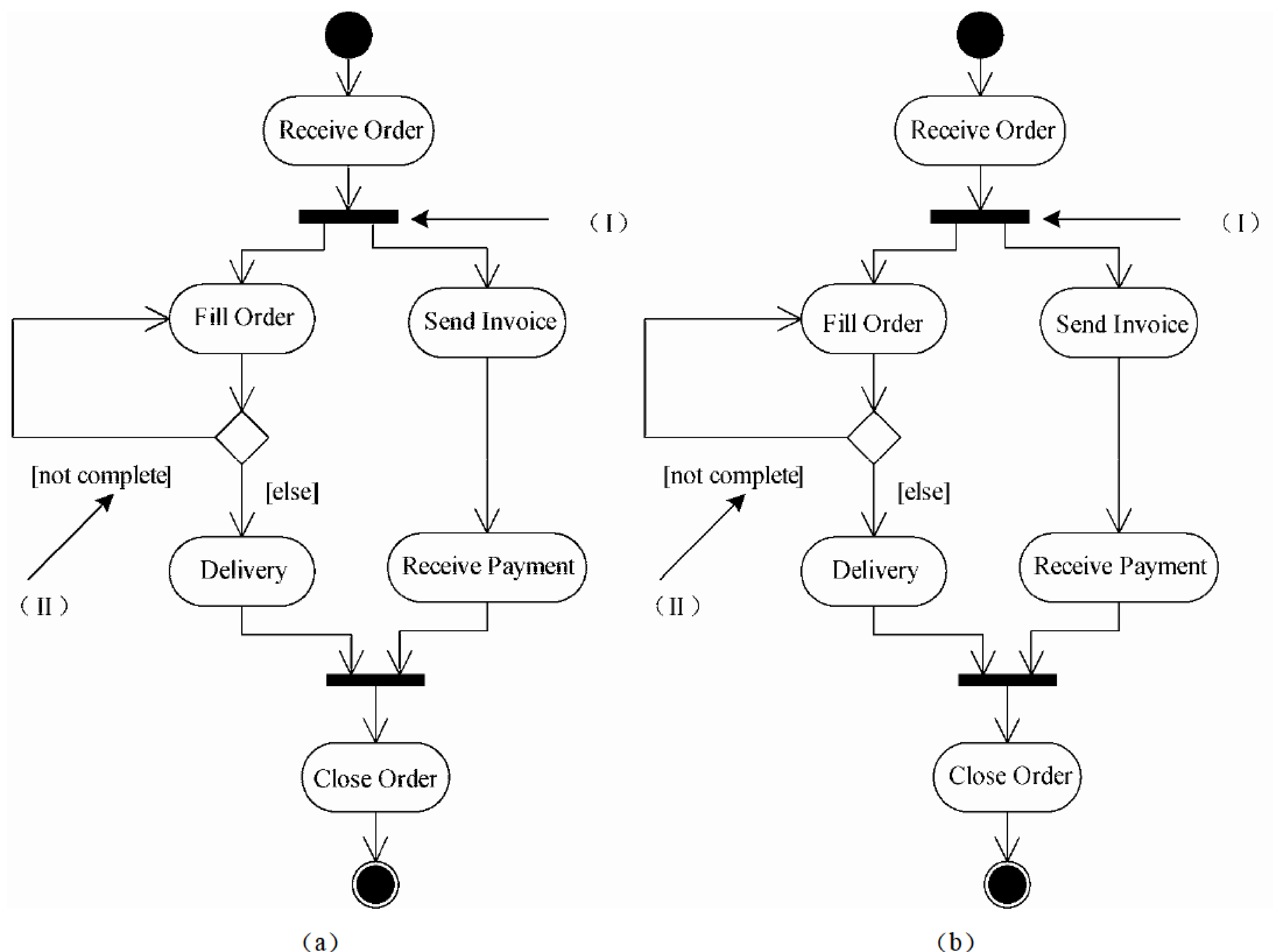
参考答案

（32）D　（33）D

试题（34）、（35）

下图（a）所示为一个模块层次结构的例子，图（b）所示为对其进行集成测试的顺

序，则此测试采用了__（34）__测试策略。该测试策略的优点不包括__（35）__。



（a）　　　（b）

（34）A. 自底向上　B. 自顶向下　C. 三明治　D. 一次性

（35）A. 较早地验证了主要的控制和判断点

B. 较早地验证了底层模块

C. 测试的并行程度较高

D. 较少的驱动模块和桩模块的编写工作量

试题（34）、（35）分析

本题考查软件测试的基础知识。

软件测试按阶段划分为单元测试、集成测试和系统测试。在单元测试基础上，将所有模块按照设计要求组装为系统，此时进行的测试称为集成测试。

集成测试有多种策略：

自底向上：从系统层次中最底层的构件开始测试，逐步向上。需要设计驱动模块来辅助测试。

自顶向下：与自底向上相反，从最顶层的构件开始，逐步向下。需要设计桩模块来辅助测试。

三明治：结合自底向上和自顶向下两种测试策略。

一次性：对所有构件一次性测试，然后集成。

根据题干，该实例采用了三明治测试策略。

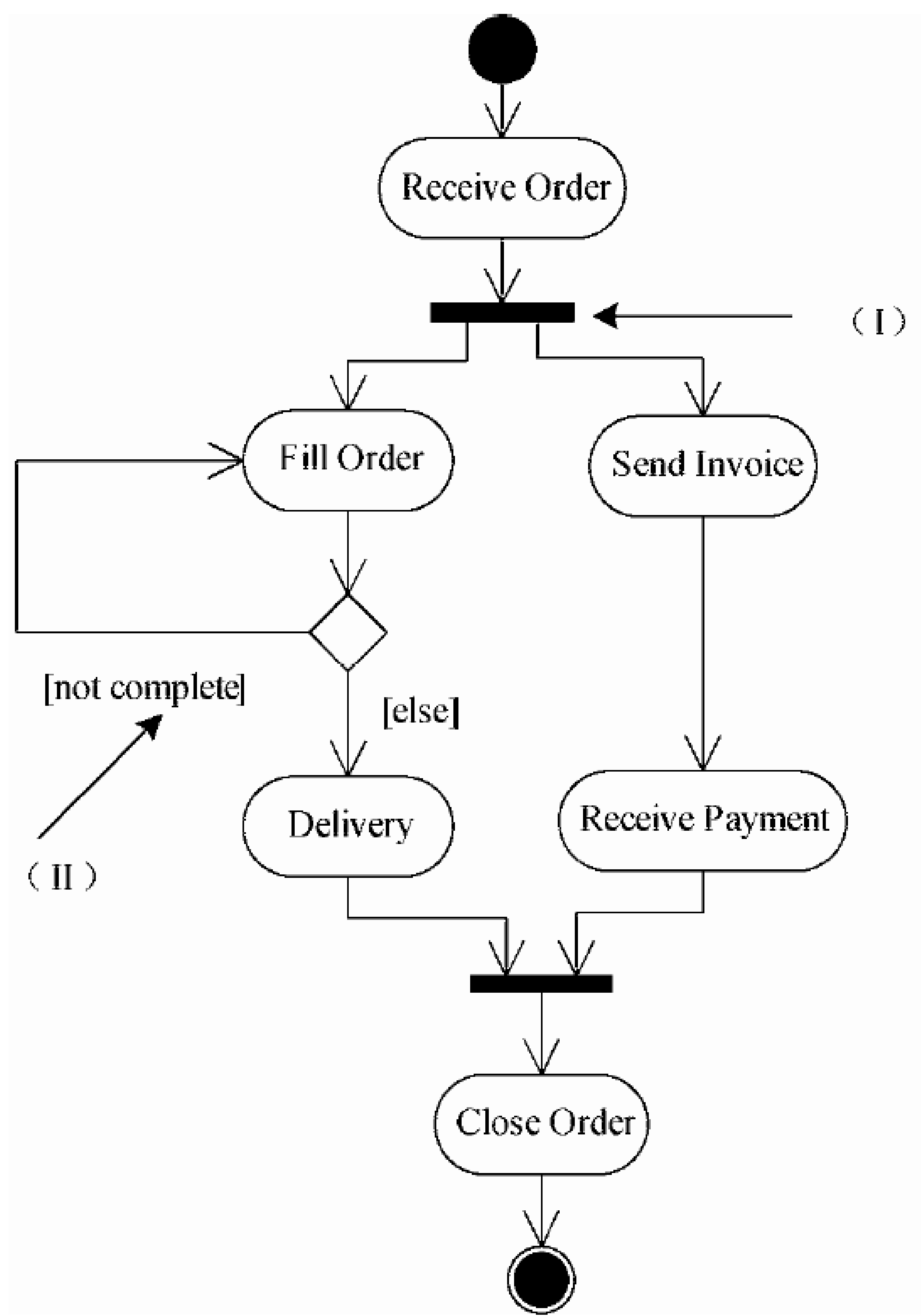
该测量的优势是结合了自底向上和自顶向下的优点，如较早地验证了主要的控制构件和底层模块，并行测试程度较高等。但缺点是需要写较多的驱动模块和桩模块。

参考答案

（34）C （35）D

试题（36）

采用 McCabe 度量法计算下图所示程序的环路复杂性为＿（36）＿。



（36）A．1　　B．2　　C．3　　D．4

试题（36）分析

本题考查软件复杂性的基础知识。

McCabe 度量法是一种基于程序控制流的复杂性度量方法，环路复杂性为 $V(G) = m - n + 2$，图中 m=11，n=10，$V(G) = 11-10+2=3$。

参考答案

（36）C

试题（37）～（38）

在面向对象方法中，<u>（37）</u>是父类和子类之间共享数据和方法的机制。子类在原有父类接口的基础上，用适合于自己要求的实现去置换父类中的相应实现称为<u>（38）</u>。

（37）A．封装　B．继承　C．覆盖　D．多态

（38）A．封装　B．继承　C．覆盖　D．多态

试题（37）～（38）分析

本题考查面向对象的基本知识。

在面向对象系统中，对象是基本的运行时实体，它既包括数据（属性），也包括作用于数据的操作（行为）。所以，一个对象把属性和行为封装为一个整体。封装是一种信息隐蔽技术，它的目的是使对象的使用者和生产者分离，使对象的定义和实现分开。一个类定义了一组大体上相似的对象。一个类所包含的方法和数据描述一组对象的共同行为和属性，这些对象共享这些行为和属性。有些类之间存在一般和特殊关系，在定义和实现一个类的时候，可以在一个已经存在的类的基础上来进行，把这个已经存在的类所定义的内容作为自己的内容，并加入新的内容，这种机制就是父类和子类之间共享数据和方法的机制，即继承。在子类定义时，可以继承它的父类（或祖先类）中的属性和方法，也可以重新定义父类中已经定义的方法，其方法可以对父类中方法进行覆盖，即在原有父类接口的基础上，用适合于自己要求的实现去置换父类中的相应实现。多态是在继承的支持下，在不同对象在收到同一消息时可以产生不同的结果，这是由于对通用消息的实现细节由接收对象自行决定的缘故。

参考答案

（37）B　（38）C

试题（39）

在 UML 用例图中，参与者表示<u>（39）</u>。

（39）A．人、硬件或其他系统可以扮演的角色

B．可以完成多种动作的相同用户

C．不管角色的实际物理用户

D．带接口的物理系统或者硬件设计

试题（39）分析

本题考查面向对象和统一建模语言（UML）的基本知识。

UML 用例图展现了一组用例、参与者（Actor）以及它们之间的关系。用于对系统的静态用例视图进行建模。这个视图主要支持以下系统的行为，即该系统在它的周边环境的语境中所提供的外部可见服务。用例图说明参与者及其扮演的角色，可以是人、硬件或者其他系统可以扮演的角色，而非个人用户。

参考答案

（39）A

试题（40）

UML 中关联是一个结构关系，描述了一组链。两个类之间__（40）__关联。

（40）A. 不能有多个　　B. 可以有多个由不同角色标识的

C. 可以有任意多个　　D. 的多个关联必须聚合成一个

试题（40）分析

本题考查面向对象和统一建模语言（UML）的基础知识。

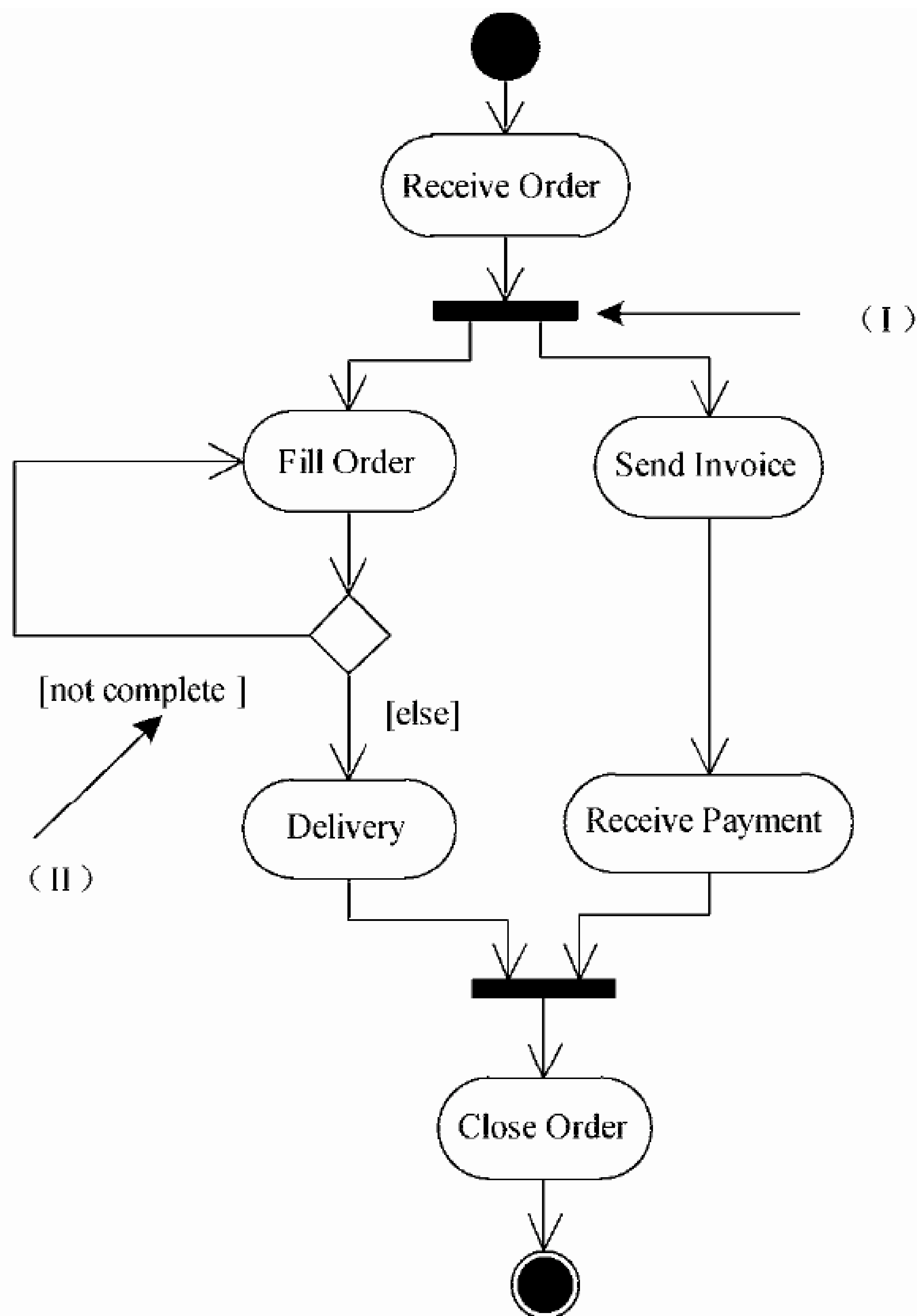
在 UML 中，关系把事物结合在一起，包括依赖、关联、泛化和实现四种。关联是一种结构关系，描述了一组链，即对象之间的连接；聚集是一种特殊类型的关联，描述了整体和部分之间的结构关系。在关联上可以标注重复度（multiplicity）和角色（role）。两个类之间可以有多个关联，但这些关联需要由不同角色进行标识。

参考答案

（40）B

试题（41）～（43）

如下所示的 UML 图是__（41）__，图中（I）表示__（42）__，（II）表示__（43）__。



（41）A. 序列图　　B. 状态图　　C. 通信图　　D. 活动图

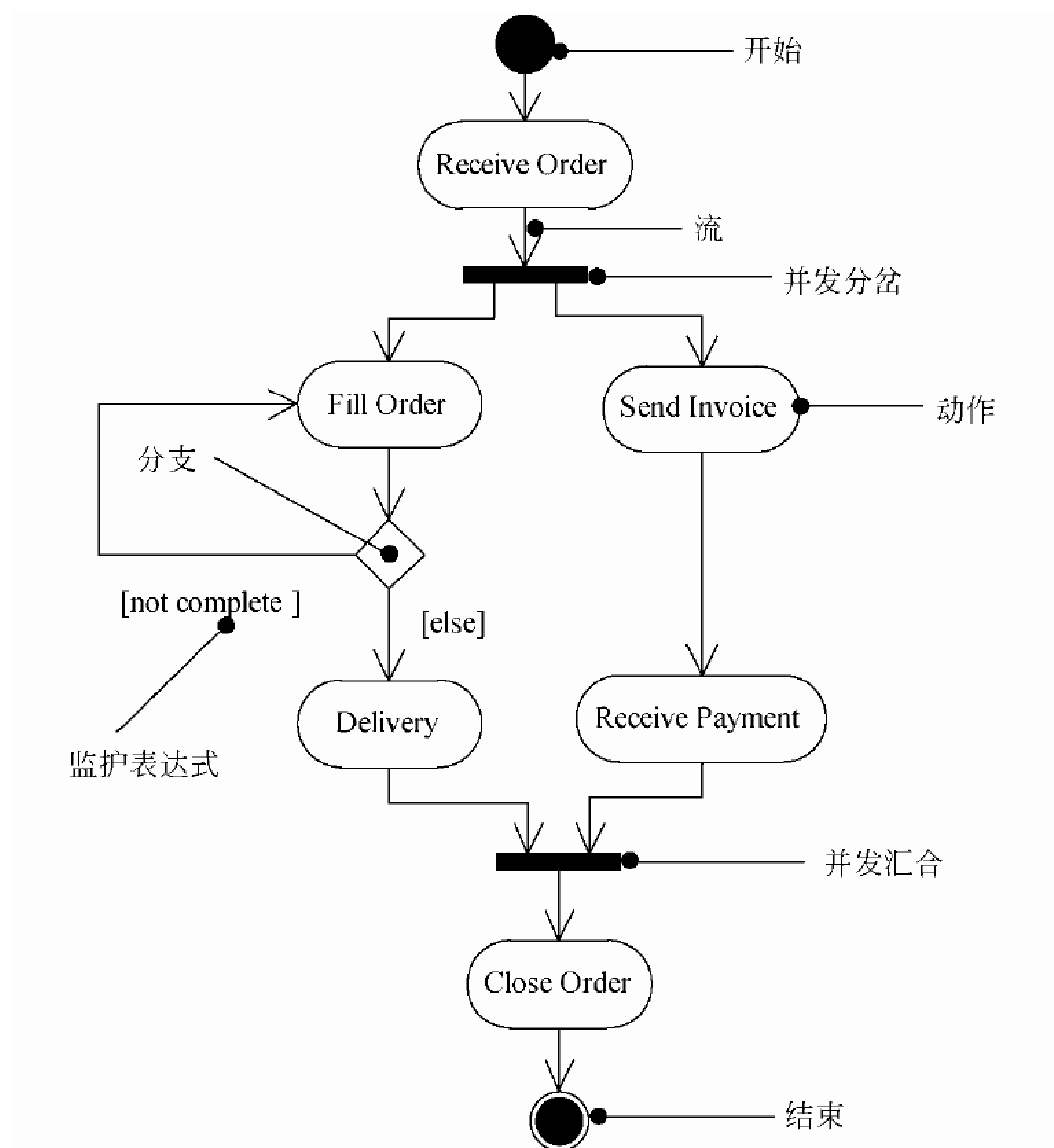
（42）A．合并分叉　　B．分支　　C．合并汇合　　D．流

（43）A．分支条件　　B．监护表达式　　C．动作名　　D．流名称

试题（41）～（43）分析

本题考查统一建模语言（UML）的基本知识。

活动图（activity diagram）是一种特殊的状态图，它展现了在系统内从一个活动到另一个活动的流程，专注于系统的动态视图，它对于系统的功能建模特别重要，并强调对象间的控制流程。如下图所示：



活动图一般包括活动状态和动作状态、转换和对象。活动图有开始、结束和一系列动作，可以表示分支、合并、分岔和汇合。分支描述基于布尔表达式的可选择路径，可有一个入流和 2 个或多个出流，在每个出流上放置一个布尔表达式条件（监护表达式），每个出流的条件不应该重叠，但需要覆盖所有可能性。合并描述当两条控制路径重新合并，不需要监护条件，只有一个出流。分岔描述把一个控制流分成两个或多个并发控制流，可以有一个进入转移和两个或多个离去转移，每个离去的转移表示一个独立的控制流，这些流可以并行的进行。汇合表示两个或多个并发控制流的同步，可以有两个或多个进入转移和一个离去转移，意味着每个进入流都等待，直到所有进入流都达到这个汇

合处。

参考答案

（41）D （42）A （43）B

试题（44）

为图形用户界面（GUI）组件定义不同平台的并行类层次结构，适合采用__（44）__模式。

（44）A．享元（Flyweight） B．抽象工厂（Abstract Factory）
C．外观（Façade） D．装饰器（Decorator）

试题（44）分析

本题考查设计模式的基本概念。每种设计模式都有特定的意图和适用情况。

享元（Flyweight）模式运用共享技术有效地支持大量细粒度的对象。适用于：一个应用程序使用了大量的对象；完全由于使用大量的对象而造成很大的存储开销；对象的大多数状态都可变为外部状态；如果删除对象的外部状态，那么可以用相对较少的共享对象取代很多组对象；应用程序不依赖于对象标识。

抽象工厂（Abstract Factory）模式提供一个创建一系列相关或相互依赖对象的接口，而无须指定它们具体的类。适用于：一个系统要独立于它的产品的创建、组合和表示时；一个系统要由多个产品系列中的一个来配置时；当要强调一系列相关的产品对象的设计以便进行联合使用时；当提供一个产品类库，而只想显示它们的接口而不是实现时。如为图形用户界面（GUI）组件定义不同平台的并行类层次结构，适合采用此模式，其中抽象工厂声明一个创建抽象界面组件的操作接口，具体工厂实现创建产品对象的操作。

外观（Façade）模式为子系统中的一组接口提供一个一致的界面，Façade 模式定义了一个高层接口，这个接口使得这一子系统更加容易使用。适用于：要为一个复杂子系统提供一个简单接口时，子系统往往因为不断演化而变得越来越复杂；客户程序与抽象类的实现部分之间存在着很大的依赖性；当需要构建一个层次结构的子系统时，使用 façade 模式定义子系统中每层的入口点。

装饰器（Decorator）模式描述了以透明围栏来支持修饰的类和对象的关系，动态地给一个对象添加一些额外的职责，从增加功能的角度来看，装饰器模式相比生成子类更加灵活。适用于：在不影响其他对象的情况下，以动态、透明的方式给单个对象添加职责；处理那些可以撤销的职责；当不能采用生成子类的方式进行扩充时。

参考答案

（44）B

试题（45）

__（45）__设计模式将一个请求封装为一个对象，从而使得可以用不同的请求对客户进行参数化，对请求排队或记录请求日志，以及支持可撤销的操作。

（45）A．命令（Command） B．责任链（Chain of Responsibility）
C．观察者（Observer） D．策略（Strategy）

试题（45）分析

本题考查设计模式的基本概念。每种设计模式都有特定的意图，描述一个在我们周围不断重复发生的问题，以及该问题的解决方案的核心，使该方案能够重用而不必做重复劳动。

命令（Command）将一个请求封装为一个对象，从而使得可以用不同的请求对客户进行参数化；对请求排队或记录请求日志，以及支持可撤销的操作。

责任链（Chain of Responsibility）使多个对象都有机会处理请求，从而避免请求的发送者和接收者之间的耦合关系。将这些对象连成一条链，并沿着这条链传递该请求，直到有一个对象处理它为止。

观察者（Observer）模式定义对象间的一种一对多的依赖关系，当一个对象的状态发生改变时，所有依赖于它的对象都得到通知并被自动更新。

策略（Strategy）定义一系列的算法，把它们一个个封装起来，并且使它们可以相互替换。此模式使得算法可以独立于使用它们的客户而变化。

参考答案

（45）A

试题（46）

（46）设计模式最适合用于发布/订阅消息模型，即当订阅者注册一个主题后，此主题有新消息到来时订阅者就会收到通知。

（46）A. 适配器（Adapter）　　B. 通知（Notifier）
C. 观察者（Observer）　　D. 状态（State）

试题（46）分析

本题考查设计模式的基本概念。每种设计模式都有特定的意图和适用情况。

适配器（Adapter）将一个类的接口转换成客户希望的另外一个接口。Adapter 模式使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。类适配使用多重继承对一个接口与另一个接口进行匹配；对象适配器依赖于对象组合。适用于：想使用一个已经存在的类，而它的接口不符合要求；想创建一个可以复用的类，该类可以与其他不相关的类或不可预见的类（即那些接口可能不一定兼容的类）协同工作。（仅适用于对象 Adapter）想使用一个已经存在的子类，但是不可能对每一个都进行子类化以匹配他们的接口。对象适配器可以适配它的父类接口。

观察者（Observer）模式定义对象间的一种一对多的依赖关系，当一个对象的状态发生改变时，所有依赖于它的对象都得到通知并被自动更新。适用于：当一个抽象模型有两个方面，其中一个方面依赖于另一个方面，将这两者封装在独立对象中以使它们可以各自独立地改变和复用；当对一个对象的改变需要同时改变其他对象，而不知道具体有多少对象有待改变时；当一个对象必须通知其他对象，而它又不能假定其他对象是谁，即：不希望这些对象是紧耦合的。此模式最适合用于发布/订阅消息模型由订阅者订阅消

息主题，发布者一定有此主题消息发布，所有订阅者就会自动收到通知。

状态（State）允许一个对象在其内部状态改变时改变它的行为。对象看起来似乎修改了它的类。适用于：一个对象的行为决定于它的状态，并且它必须在运行时刻根据状态改变它的行为；一个操作中含有庞大的多分支的条件语句，且这些分支依赖于该对象的状态。这个状态通常用一个或多个枚举常量表示。通常，有多个操作包含这一相同的条件结构。State 模式将每一个条件分支放入一个独立的类中。这使得开发者可以根据对象自身的情况将对象的状态作为一个对象，这一对象可以不依赖于其他对象而独立变化。

参考答案

（46）C

试题（47）

因使用大量的对象而造成很大的存储开销时，适合采用 （47） 模式进行对象共享，以减少对象数量从而达到较少的内存占用并提升性能。

（47）A. 组合（Composite） B. 享元（Flyweight）
C. 迭代器（Iterator） D. 备忘（Memento）

试题（47）分析

本题考查设计模式的基本概念。每种设计模式都有特定的意图和适用情况。

组合（Composite）模式将对象组合成树形结构以表示“部分-整体”的层次结构，使得用户对单个对象和组合对象的使用具有一致性。组件 Component 为组合的对象声明接口，通常定义父组件引用，用户引用此组件，Leaf 和 Composite 类可以继承这个引用以及管理这个应用的那些操作。适用于：想表示对象的“部分-整体”层次结构；希望用户忽略组合对象与单个对象的不同，用户将统一地使用组合结构中的所有对象。

享元（Flyweight）模式运用共享技术有效地支持大量细粒度的对象。

迭代器（Iterator）提供一种方法顺序访问一个聚合对象中各个元素，而又不需暴露该对象的内部表示。适用于：访问一个聚合对象的内容而无须暴露它的内部表示；支持对聚合对象的多种遍历；为遍历不同的聚合结构提供一个统一的接口。

备忘（Memento）在不破坏封装性的前提下，捕获一个对象的内部状态，并在对象之外保存这个状态。这样以后就可将对象恢复到原先保存的状态。适用于：必须保存一个对象在某一个时刻的（部分）状态，这样以后需要时它才能恢复到先前的状态；如果一个用接口来让其他对象直接得到这些状态，将会暴露对象的实现细节并破坏对象的封装性。

参考答案

（47）B

试题（48）

移进—归约分析法是编译程序（或解释程序）对高级语言源程序进行语法分析的一种方法，属于 （48） 的语法分析方法。

（48）A．自顶向下（或自上而下）　　B．自底向上（或自下而上）
　　　C．自左向右　　D．自右向左

试题（48）分析

本题考查程序语言基础知识。

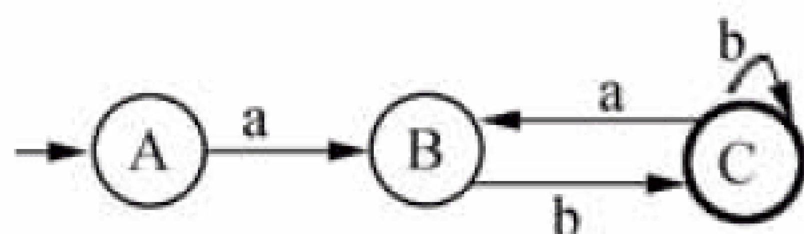
语法分析方法分为两类：自上而下（自顶向下）分析法和自下而上（自底向上）分析法，递归下降分析法和预测分析法属于自上而下分析法，移进-归约分析法属于自下而上（自底向上）分析法。

参考答案

（48）B

试题（49）

某确定的有限自动机（DFA）的状态转换图如下图所示（A 是初态，C 是终态），则该 DFA 能识别＿（49）＿。



（49）A．aabb　　B．abab　　C．baba　　D．abba

试题（49）分析

本题考查程序语言基础知识。

在 DFA 中，如果存在从初态到达终态的路径，其上的标记字母构成字符串 s，则称该 DFA 可以识别 s。

根据题目中的状态转换图，对于 aabb，从状态 A 出发，识别字母“a”后转到状 B，接下来不存在字母“a”的状态转换，因此，该 DFA 不能识别 aabb。

对于 abab，其识别路径为 A->B->C->B->C，当字符串结束时，到达终态 C，因此该 DFA 能识别 abab。

对于 baba，不存在识别路径，因为从状 A 出发没有字母“b”的状态转换。

对于 abba，其识别路径为 A->B->C->C->B，字符串结束时不在终态，因此该 DFA 不能识别 abba。

参考答案

（49）B

试题（50）

函数 main()、f()的定义如下所示，调用函数 f()时，第一个参数采用传值（call by value）方式，第二个参数采用传引用（call by reference）方式，main 函数中“print(x)”执行后输出的值为＿（50）＿。

```
main( )
int x = 1;
f(5, x);
print(x);
```

```
f(int x, int &a)
x = 2*x +1 ;
a = a + x;
return;
```

（50）A. 1　　　　B. 6　　　　C. 11　　　　D. 12

试题（50）分析

本题考查程序语言基础知识。

函数调用执行时，传值调用是指将实参的值传给形参，形参变量得到实参值的一份拷贝，引用调用实质上将实参变量的地址传给形参变量，在被调用函数中通过指针间接访问实参变量，这样，对形参的修改实质上是对实参变量的修改。本题中，函数 f 执行时，其形参 x 得到的值为 5，语句“x=2*x+1;”将函数 f 中 x 的值改变为 11，而形参 a 实质上引用的是 main 函数中的 x，若用 main_x 表示面函数中的变量 x，用 f_x 表示函数 f 中的形参变量 x，则语句“a=a+x;”的实质是“main_x = main_x + f_x;”，因此结果是 main 函数中 x 的值改为 12。

参考答案

（50）D

试题（51）

数据的物理独立性和逻辑独立性分别是通过修改__（51）__来完成的。

（51）A. 外模式与内模式之间的映像、模式与内模式之间的映像
　　B. 外模式与内模式之间的映像、外模式与模式之间的映像
　　C. 外模式与模式之间的映像、模式与内模式之间的映像
　　D. 模式与内模式之间的映像、外模式与模式之间的映像

试题（51）分析

本题考查数据库基本知识。

数据的独立性是由 DBMS 的二级映像功能来保证的。数据的独立性包括数据的物理独立性和数据的逻辑独立性。数据的物理独立性是指当数据库的内模式发生改变时，数据的逻辑结构不变。为了保证应用程序能够正确执行，需要通过修改概念模式/内模式之间的映像。数据的逻辑独立性是指用户的应用程序与数据库的逻辑结构是相互独立的。数据的逻辑结构发生变化后，用户程序也可以不修改。但是，为了保证应用程序能够正确执行，需要修改外模式/概念模式之间的映像。

参考答案

（51）D

试题（52）

关系规范化在数据库设计的__（52）__阶段进行。

（52）A. 需求分析　　B. 概念设计　　C. 逻辑设计　　D. 物理设计

试题（52）分析

逻辑设计阶段的任务之一是对关系模式进一步的规范化处理。因为生成的初始关系模式并不能完全符合要求，还会有数据冗余、更新异常存在，这就需要根据规范化理论对关系模式分解之后来消除。不过有时根据处理要求，可能还需要增加部分冗余以满足处理要求。逻辑设计阶段的任务就需要作部分关系模式的处理，分解、合并或增加冗余属性，提高存储效率和处理效率。

参考答案

（52）C

试题（53）

若给定的关系模式为 $R<U,F>$，$U=\{A,B,C\}$，$F=\{AB\to C,C\to B\}$，则关系 R ＿（53）＿。

（53）A. 有 2 个候选关键字 AC 和 BC，并且有 3 个主属性

B. 有 2 个候选关键字 AC 和 AB，并且有 3 个主属性

C. 只有 1 个候选关键字 AC，并且有 1 个非主属性和 2 个主属性

D. 只有 1 个候选关键字 AB，并且有 1 个非主属性和 2 个主属性

试题（53）分析

本题考查关系数据库规范化理论方面的基础知识。

根据函数依赖定义可知 AC→U 和 AB→U，所以 AC 和 AB 为候选关键字。根据主属性的定义“包含在任何一个候选码中的属性叫做**主属性**（Prime attribute），否则叫做**非主属性**（Nonprime attribute）”，所以，关系 R 中的 3 个属性都是主属性。

参考答案

（53）B

试题（54）～（56）

某公司数据库中的元件关系模式为 P（元件号，元件名称，供应商，供应商所在地，库存量），函数依赖集 F 如下所示：

F＝{元件号→元件名称，（元件号，供应商）→库存量，供应商→供应商所在地}

元件关系的主键为＿（54）＿，该关系存在冗余以及插入异常和删除异常等问题。为了解决这一问题需要将元件关系分解为＿（55）＿，分解后的关系模式可以达到＿（56）＿。

（54）A. 元件号，　元件名称　　B. 元件号，供应商

C. 元件号，供应商所在地　　D. 供应商，供应商所在地

（55）A. 元件 1（元件号，元件名称，库存量）、元件 2（供应商，供应商所在地）

B. 元件 1（元件号，元件名称）、元件 2（供应商，供应商所在地，库存量）

C. 元件 1（元件号，元件名称）、元件 2（元件号，供应商，库存量）、元件 3（供应商，供应商所在地）

D. 元件 1（元件号，元件名称）、元件 2（元件号，库存量）、元件 3（供应

商，供应商所在地）、元件 4（供应商所在地，库存量）

（56）A. 1NF　　B. 2NF　　C. 3NF　　D. 4NF

试题（54）～（56）分析

试题（54）的正确选项为 B。由于（元件号，供应商）可以决定全属性，即（元件号，供应商）→元件名称，供应商所在地，库存量，所以元件关系的主键为（元件号，供应商）。

试题（55）的正确选项为 C。因为关系 P 存在冗余以及插入异常和删除异常等问题，为了解决这一问题需要将元件关系分解。选项 A、选项 B 和选项 D 是有损连接的，且不保持函数依赖故分解是错误的，例如，分解为选项 A、选项 B 和选项 D 后，用户无法查询某元件是由哪些供应商来供应，原因是分解有损连接的，且不保持函数依赖。

试题（56）的正确选项为 C。因为原元件关系存在非主属性对码的部分函数依赖：（元件号，供应商）→供应商所在地，但是供应商→供应商所在地，故原关系模式元件非 2NF 的。分解后的关系模式元件 1、元件 2 和元件 3 消除了非主属性对码的部分函数依赖，同时不存在传递依赖，故达到 3NF。

参考答案

（54）B　（55）C　（56）C

试题（57）

若元素以 a,b,c,d,e 的顺序进入一个初始为空的栈中，每个元素进栈、出栈各 1 次，要求出栈的第一个元素为 d，则合法的出栈序列共有__（57）__种。

（57）A. 4　　B. 5　　C. 6　　D. 24

试题（57）分析

本题考查数据结构基础知识。

栈的修改规则是后进先出。对于题目给出的元素序列，若要求 d 先出栈，则此时 a、b、c 尚在栈中，因此这四个元素构成的出栈序列只能是 d c b a，元素 e 可在 c 出栈之前进栈，之后 c 也只能在 e 出栈后再出栈，因此可以得到出栈系列 d e c b a。同理，e 可在 b 出栈之前进栈，从而得到出栈序列 d c e b a。若 e 在 a 出栈前入栈，则得到出栈序列 d c b e a，或者 e 在 a 出栈后进、出栈，从而得到出栈序列 d c b a e。

参考答案

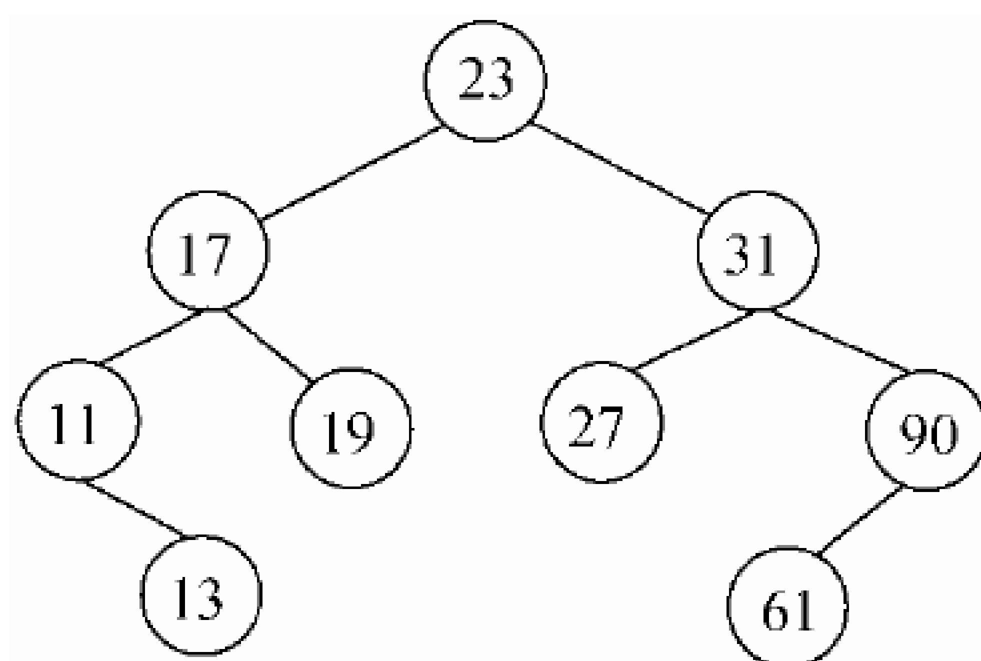
（57）A

试题（58）

设有二叉排序树（或二叉查找树）如下图所示，建立该二叉树的关键码序列不可能是__（58）__。

（58）A. 23 31 17 19 11 27 13 90 61　　B. 23 17 19 31 27 90 61 11 13

C. 23 17 27 19 31 13 11 90 61　　D. 23 31 90 61 27 17 19 11 13



试题（58）分析

本题考查数据结构基础知识。

根据二叉排序树的定义，将新结点插入二叉排序树时，需要先查找插入位置。若等于树根，则不再插入，若大于树根，则递归地在右子树上查找插入位置，否则递归地在左子树上查找插入位置，因此，新结点总是以叶子的方式加入树中。这样，在根结点到达每个叶子结点的路径上，结点的顺序必须保持，也就是父结点必定先于子结点进入树中。

考查题目中的序列，在序列“23 17 27 19 31 13 11 90 61”中，27 先于 31 进入该二叉排序树，这是不可能的。

参考答案

（58）C

试题（59）

若一棵二叉树的高度（即层数）为 h，则该二叉树＿（59）＿。

（59）A．有 2^h 个结点　　B．有 2^h-1 个结点
　　C．最少有 2^h-1 个结点　　D．最多有 2^h-1 个结点

试题（59）分析

本题考查数据结构基础知识。

二叉树中，非叶子结点最多有两个子结点，第 i 层上最多有 $2^{(i-1)}$ 个结点，因此高度为 h 的二叉树最多有 2^h-1 个结点。

参考答案

（59）D

试题（60）

在 13 个元素构成的有序表 A[1..13]中进行折半查找（或称为二分查找，向下取整）。那么以下叙述中，错误的是＿（60）＿。

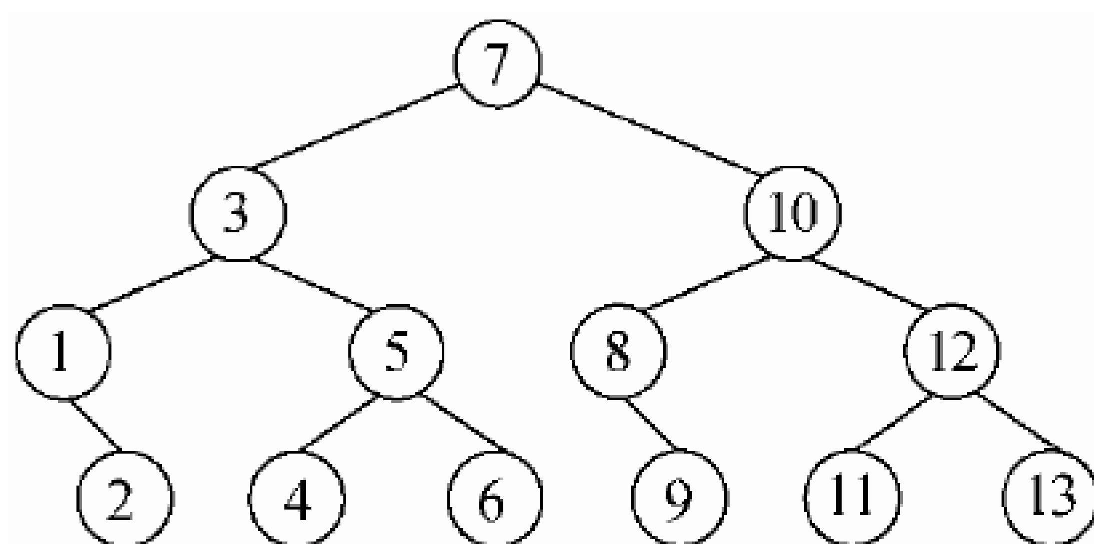
（60）A．无论要查找哪个元素，都是先与 A[7]进行比较
B．若要查找的元素等于 A[9]，则分别需与 A[7]、A[11]、A[9]进行比较
C．无论要查找的元素是否在 A[]中，最多与表中的 4 个元素比较即可
D．若待查找的元素不在 A[]中，最少需要与表中的 3 个元素进行比较

试题（60）分析

本题考查数据结构基础知识。

设查找表的元素存储在一维数组 r[1..n]中，在表中的元素已经按关键字递增方式排序的情况下，进行折半查找的方法是：首先将待查元素的关键字（key）值与数组 r 中间位置上（下标为 mid）记录的关键字进行比较，若相等，则查找成功；若 key>r[mid].key，则说明待查记录只可能在后半个子表 r[mid+1..n]中，下一步应在后半个子表中进行查找，若 key<r[mid].key，说明待查记录只可能在前半个子表 r[1..mid–1]中，下一步应在 r 的前半个子表中进行查找，这样就可以迅速逐步缩小范围，直到查找成功或子表为空时失败为止。

折半查找过程可用一棵二叉树表示，其中结点中的数字表示元素的下标。



参考答案

（60）B

试题（61）

以下关于图的遍历的叙述中，正确的是__（61）__。

（61）A. 图的遍历是从给定的源点出发对每一个顶点仅访问一次的过程

B. 图的深度优先遍历方法不适用于无向图

C. 使用队列对图进行广度优先遍历

D. 图中有回路时则无法进行遍历

试题（61）分析

本题考查数据结构基础知识。

图的遍历是指对图中所有顶点进行访问且只访问一次的过程。因为图的任一个结点都可能与其余顶点相邻接，所以在访问了某个顶点之后，可能沿着某路径又回到该结点上。因此为了避免顶点的重复访问，在图的遍历过程中，必须对已访问过的顶点进行标记。深度优先遍历和广度优先遍历是两种遍历图的基本方法。

图的广度优先遍历方法为：从图中某个顶点 *v* 出发，在访问了 *v* 之后依次访问 *v* 的各个未被访问过的邻接点，然后分别从这些邻接点出发依次访问它们的邻接点，并使“先被访问的顶点的邻接点”先于“后被访问的顶点的邻接点”被访问，直至图中所有已被访问的顶点的邻接点都被访问到。若此时还有未被访问的顶点，则另选图中的一个未被访问的顶点作为起点，重复上述过程，直至图中所有的顶点都被访问到为止。

广度优先遍历图的特点是尽可能先进行横向搜索，即最先访问的顶点的邻接点也先被访问。为此，引入队列来保存已访问过的顶点序列，即每当一个顶点被访问后，就将其放入队中，当队头顶点出队时，就访问其未被访问的邻接点并令这些邻接顶点入队。

参考答案

（61）C

试题（62）～（65）

考虑一个背包问题，共有 $n = 5$ 个物品，背包容量为 $W = 10$，物品的重量和价值分别为：$w = \{2, 2, 6, 5, 4\}$，$v = \{6, 3, 5, 4, 6\}$，求背包问题的最大装包价值。若此为 0-1 背包问题，分析该问题具有最优子结构，定义递归式为

$$c[i,j]=\begin{cases} 0 & \text{若} i=0\text{或} j=0 \\ c[i-1,j] & \text{若} w[i]>j \\ \max\{c(i-1,j)+c(i-1,j-w(i))\} & \text{其他} \end{cases}$$

其中 $c(i, j)$表示 i 个物品、容量为 j 的 0-1 背包问题的最大装包价值，最终要求解 $c(n, W)$。

采用自底向上的动态规划方法求解，得到最大装包价值为 （62） ，算法的时间复杂度为 （63） 。

若此为部分背包问题，首先采用归并排序算法，根据物品的单位重量价值从大到小排序，然后依次将物品放入背包直至所有物品放入背包中或者背包再无容量，则得到的最大装包价值为 （64） ，算法的时间复杂度为 （65） 。

（62）A. 11　　B. 14　　C. 15　　D. 16.67

（63）A. $\Theta(nW)$　　B. $\Theta(n\lg n)$　　C. $\Theta(n^2)$　　D. $\Theta(n\lg nW)$

（64）A. 11　　B. 14　　C. 15　　D. 16.67

（65）A. $\Theta(nW)$　　B. $\Theta(n\lg n)$　　C. $\Theta(n^2)$　　D. $\Theta(n\lg nW)$

试题（62）、（65）分析

本题考查算法设计与分析的基础知识。

背包问题是一个经典的计算问题，有很多应用。背包问题有两类，0-1 背包问题和部分背包问题。

若用 $c(i, j)$表示 i 个物品、容量为 j 的最大装包价值，则 0-1 背包问题可以用动态规划方法求解，其递归式为：

$$c[i,j]=\begin{cases} 0 & \text{若} i=0\text{或} j=0 \\ c[i-1,j] & \text{若} w[i]>j \\ \max\{c[i-1,j-w[i]]+v[i],c[i-1,j]\} & \text{其他} \end{cases}$$

根据该递归式，自底向上可以计算题干实例中各个子问题的最优解的值，如下表所示。

		0	1	2	3	4	5	6	7	8	9	10
wi	vi	0	0	0	0	0	0	0	0	0	0	0
2	6	0	0	6	6	6	6	6	6	6	6	6
2	3	0	0	6	6	9	9	9	9	9	9	9
6	5	0	0	6	6	9	9	9	9	11	11	14
5	4	0	0	6	6	9	9	9	10	11	13	14
4	6	0	0	6	6	9	9	12	12	15	15	15

上表中行表示物品，列表示背包容量，每个元素的值表示，在仅考虑前 i 个物品时，背包容量为该列对应的值时，所获得的最大价值。

根据上表的结果，得到最大价值为 15。

自底向上计算该递归式，在实现时其实是两重循环，物品个数的循环和背包容量的循环，因此时间复杂度为 $\Theta(nW)$。

部分背包问题可以用贪心算法求解。首先根据物品的单位重量价值对物品并对其从大到小排序，然后依次取出物品放入背包，直到所有物品装完或者背包不能装入某个物品时，只放入该物品的一部分，让背包装满。单位重量价值如下表。

	1	2	3	4	5
w	2	2	6	5	4
v	6	3	5	4	6
v/w	3	1.5	0.83	0.8	1.5

上表中行表示物品信息，即重量，价值和单位重量价值，列表示对应的物品。

根据贪心策略，首先取出第一个物品放入背包，然后取出第二个物品和第五个物品放入背包，此时获得价值 6+3+6=15，背包剩余容量 10–2–2–4=8。此时不能将第三个物品全部放入背包，只能放 2/6=1/3，对应获得的价值为 5*1/3=1.67，因此得到所获得的最大价值为 15+1.67=16.67。

若用时间复杂度为 $\Theta(n\lg n)$的归并排序算法先对物品的单位重量价值排序，然后依次将物品放入背包（时间复杂度为 $\Theta(n)$），则整个算法的时间复杂度为 $\Theta(n\lg n)$。

参考答案

（62）C （63）A （64）D （65）B

试题（66）、（67）

默认情况下，FTP 服务器的控制端口为<u>（66）</u>，上传文件时的端口为<u>（67）</u>。

（66）A．大于 1024 的端口 B．20 C．80 D．21

（67）A．大于 1024 的端口 B．20 C．80 D．21

试题（66）、（67）分析

本题考查 FTP 协议的基础知识。

默认情况下，FTP 服务器的控制端口为 21，数据端口为 20。

参考答案

（66）D　（67）B

试题（68）

使用 ping 命令可以进行网络检测，在进行一系列检测时，按照由近及远原则，首先执行的是__（68）__。

（68）A．ping 默认网关　B．ping 本地 IP　C．ping 127.0.0.1　D．ping 远程主机

试题（68）分析

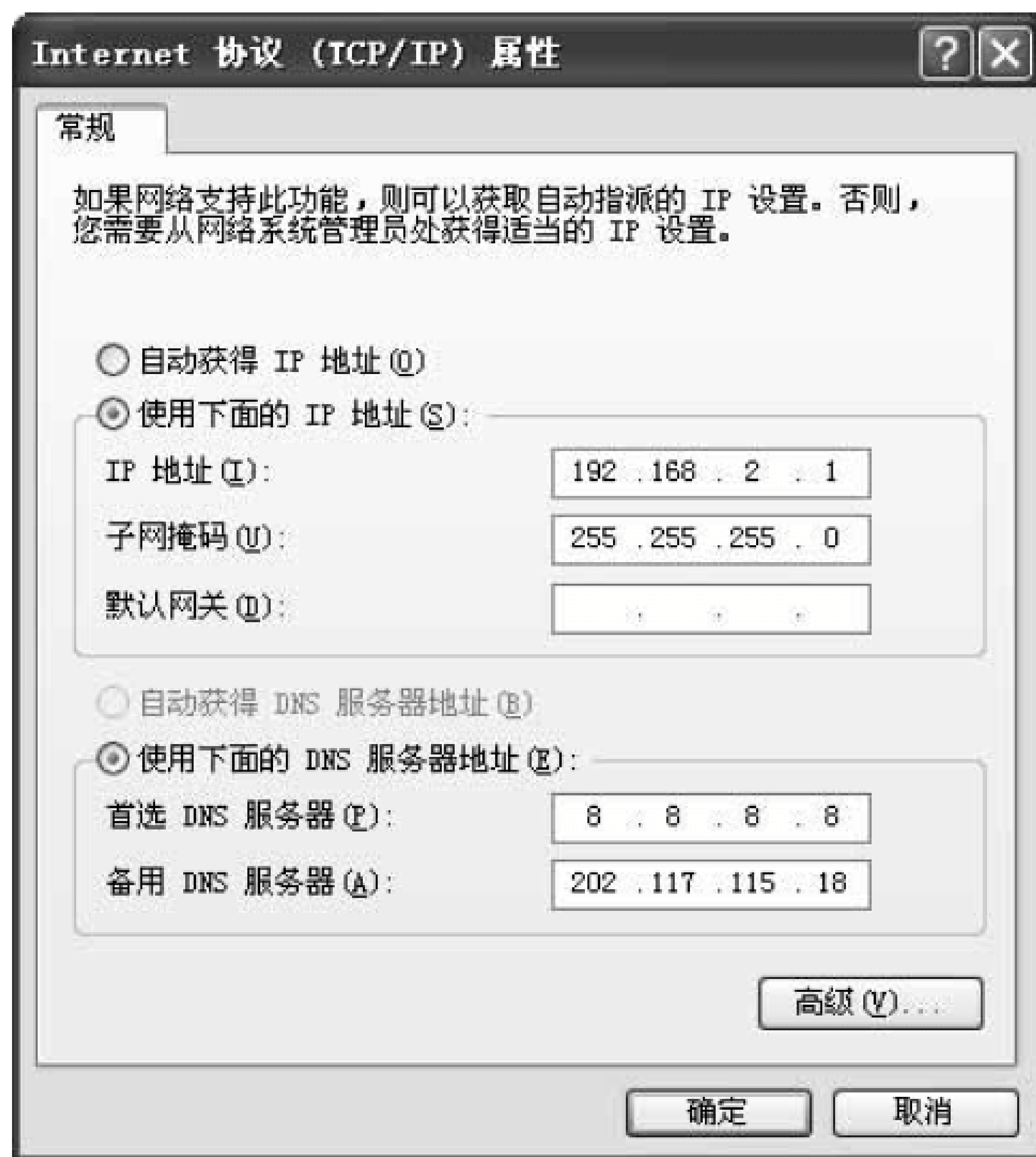
使用 ping 命令进行网络检测，按照由近及远原则，首先执行的是 ping 127.0.0.1，其次是 ping 本地 IP，再次是 ping 默认网关，最后是 ping 远程主机。

参考答案

（68）C

试题（69）

某 PC 的 Internet 协议属性参数如下图所示，默认网关的 IP 地址是__（69）__。



（69）A．8.8.8.8　B．202.117.115.3　C．192.168.2.254　D．202.117.115.18

试题（69）分析

本题考查 Internet 协议属性参数的配置。

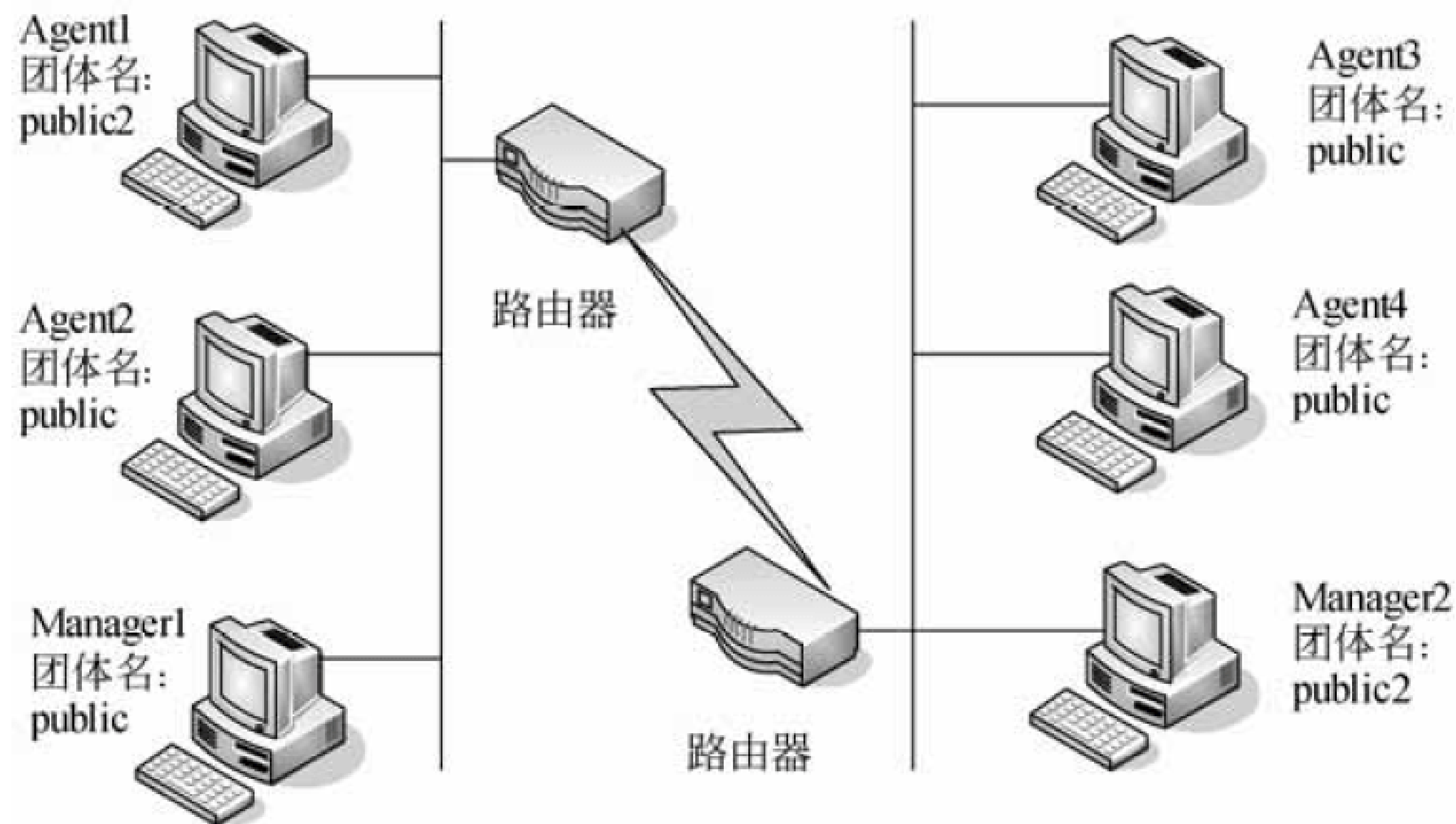
默认网关和本地 IP 地址应属同一网段。

参考答案

（69）C

试题（70）

在下图的 SNMP 配置中，能够响应 Manager2 的 getRequest 请求的是__（70）__。



（70）A．Agent1　　B．Agent2　　C．Agent3　　D．Agent4

试题（70）分析

在 SNMP 管理中，管理站和代理之间进行信息交换时要通过团体名认证，这是一种简单的安全机制，管理站与代理必须具有相同的团体名才能互相通信。但是由于包含团体名的 SNMP 报文是明文传送，所以这样的认证机制是不够安全的。本题中的 Manager2 和 Agent1 的团体名都是 public2，所以二者可以互相通信。

参考答案

（70）A

试题（71）～（75）

In the fields of physical security and information security, access control is the selective restriction of access to a place or other resource. The act of accessing may mean consuming, entering, or using. Permission to access a resource is called authorization (授权).

An access control mechanism ＿（71）＿ between a user (or a process executing on behalf of a user) and system resources, such as applications, operating systems, firewalls, routers, files, and databases. The system must first authenticate (验证) a user seeking access. Typically the authentication function determines whether the user is ＿（72）＿ to access the system at all. Then the access control function determines if the specific requested access by this user is permitteD. A security administrator maintains an authorization database that specifies what type of access to which resources is allowed for this user. The access control function consults this database to determine whether to ＿（73）＿ access. An auditing function monitors and keeps a record of user accesses to system resources.

In practice, a number of ＿（74）＿ may cooperatively share the access control function. All operating systems have at least a rudimentary (基本的), and in many cases a quite robust, access control component. Add-on security packages can add to the ＿（75）＿ access control

capabilities of the OS. Particular applications or utilities, such as a database management system, also incorporate access control functions. External devices, such as firewalls, can also provide access control services.

（71）A．cooperates　B．coordinates　C．connects　D．mediates
（72）A．denied　B．permitted　C．prohibited　D．rejected
（73）A．open　B．monitor　C．grant　D．seek
（74）A．components　B．users　C．mechanisms　D．algorithms
（75）A．remote　B．native　C．controlled　D．automated

参考译文

在物理安全和信息安全领域，访问控制是访问一个地方或其他资源的选择性限制。访问的行为可能是消耗、进入或使用。访问资源的权限称为授权。

访问控制机制介于用户（或代表用户的过程的执行）和系统资源之间，资源如应用程序、操作系统、防火墙、路由器、文件和数据库。系统必须首先认证用户的访问企图。典型的，认证功能确定一个用户是否被允许访问该系统。然后，访问控制功能确定此用户的特定访问请求是否允许。安全管理员维护授权数据库，其中指定用户可以访问对那个资源具有什么类型的访问权限。访问控制功能查询数据库以确定是否授权访问。审计功能监控和记录用户对系统资源的访问。

实际上，很多组件可以一起合作提供访问控制功能。所有操作系统至少具有基本的访问控制组件，而且这些组件大多情况下非常健壮。附加安全包可以添加到操作系统的本地安全控制功能。特定的应用和实用工具，如数据管理系统，也并入了访问控制功能。如防火墙等外部设备也能够提供访问控制服务。

参考答案

（71）D　（72）B　（73）C　（74）A　（75）B

第 30 章　2016 上半年软件设计师下午试题分析与解答

试题一（共 15 分）

阅读下列说明和图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

【说明】

某会议中心提供举办会议的场地设施和各种设备，供公司与各类组织机构租用。场地包括一个大型报告厅、一个小型报告厅以及诸多会议室。这些报告厅和会议室可提供的设备有投影仪、白板、视频播放/回放设备、计算机等。为了加强管理，该中心欲开发一会议预订系统，系统的主要功能如下。

（1）检查可用性。客户提交预订请求后，检查预订表，判定所申请的场地是否在申请日期内可用；如果不可用，返回不可用信息。

（2）临时预订。会议中心管理员收到客户预定请求的通知之后，提交确认。系统生成新临时预订存入预订表，并对新客户创建一条客户信息记录加以保存。根据客户记录给客户发送临时预订确认信息和支付定金要求。

（3）分配设施与设备。根据临时预订或变更预定的设备和设施需求，分配所需设备（均能满足用户要求）和设施，更新相应的表和预订表。

（4）确认预订。管理员收到客户支付定金的通知后，检查确认，更新预订表，根据客户记录给客户发送预订确认信息。

（5）变更预订。客户还可以在支付余款前提交变更预订请求，对变更的预订请求检查可用性，如果可用，分配设施和设备；如果不可用，返回不可用信息。管理员确认变更后，根据客户记录给客户发送确认信息。

（6）要求付款。管理员从预订表中查询距预订的会议时间两周内的预定，根据客户记录给满足条件的客户发送支付余款要求。

（7）支付余款。管理员收到客户余款支付的通知后，检查确认，更新预订表中的已支付余款信息。

现采用结构化方法对会议预定系统进行分析与设计，获得如图 1-1 所示的上下文数据流图和图 1-2 所示的 0 层数据流图（不完整）。

【问题 1】（2 分）

使用说明中的词语，给出图 1-1 中的实体 E1～E2 的名称。

【问题 2】（4 分）

使用说明中的词语，给出图 1-2 中的数据存储 D1～D4 的名称。

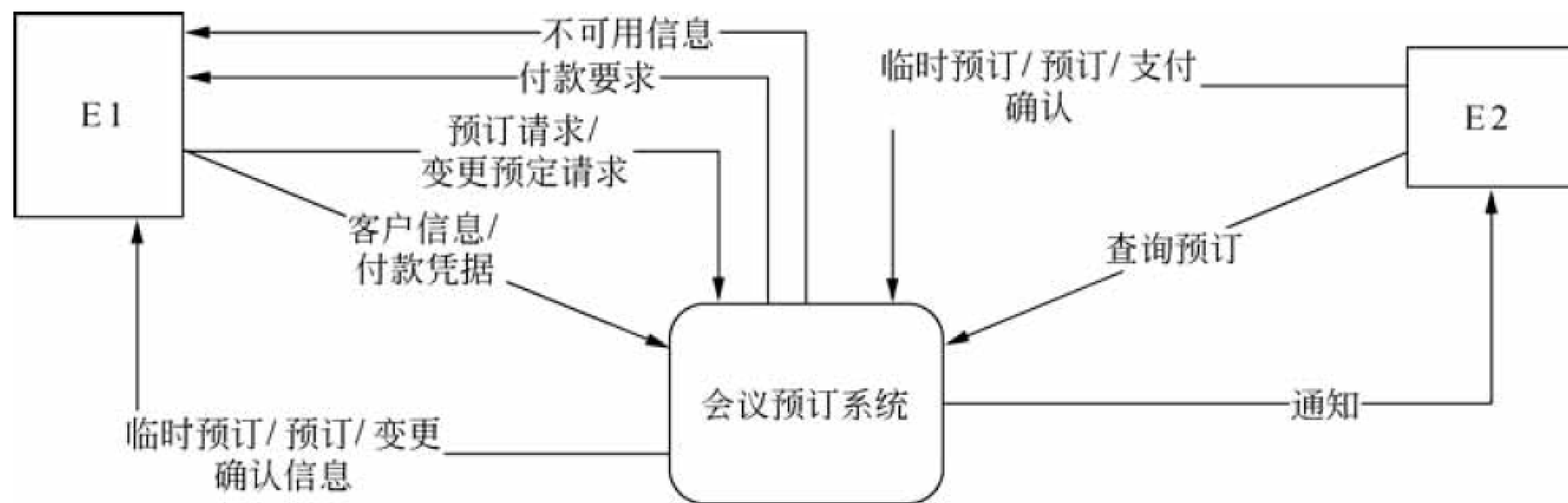


图 1-1　上下文数据流图

E1
临时预订确认信息+定金要求
2
临时预订
新客户信息
客户记录
D2
预定请求通知
临时预订确认
E2
不可用信息
可预订请求
预订请求
已支付定金凭据
1
检查可用性
设备和设施需求
新临时预订
3
分配设施和设备
4
确认预订
支付确认信息
DI
设备和设施需求
D4
D3
1
检查可用性
已支付余款信息
变更的预订请求
客户支付定金通知
预订确认
7
支付余款
5
变更预订
变更预订确认
E2
变更确认信息
变更预订请求
余款支付确认
客户余款支付通知
E2
查询预订
6
要求付款
支付余款要求
E1

图 1-2　0 层数据流图

【问题 3】（6 分）

根据说明和图中术语，补充图 1-2 中缺失的数据流及其起点和终点。

【问题 4】（3 分）

如果发送给客户的确认信息是通过 Email 系统向客户信息中的电子邮件地址进行发送的，那么需要对图 1-1 和图 1-2 进行哪些修改？用 150 字以内文字加以说明。

试题一分析

本题考查采用结构化方法进行系统分析与设计，主要考查数据流图（DFD）的应用，是比较传统的题目，考点与往年类似，要求考生细心分析题目中所描述的内容。

面向数据流建模是目前仍然被广泛使用的结构化分析与设计的方法之一，而 DFD 是面向数据流建模的重要工具，是一种便于用户理解、分析系统数据流程的图形化建模工具，是系统逻辑模型的重要组成部分。DFD 将系统建模成“输入—加工（处理）—输出”的模型，即流入软件的数据对象、经由加工的转换、最后以结果数据对象的形式流出软件，并采用分层的方式加以表示。

上下文 DFD（顶层 DFD）通常用来确定系统边界，将待开发系统看作一个大的加工（处理），然后根据系统从哪些外部实体接收数据流，以及系统将数据流发送到哪些外部实体，建模出的上下文图中只有唯一的一个加工和一些外部实体，以及这两者之间的输入输出数据流。0 层 DFD 在上下文确定的系统外部实体以及与外部实体的输入输出数据流的基础上，将上下文 DFD 中的加工分解成多个加工，识别这些加工的输入输出数据流，使得所有上下文 DFD 中的输入数据流，经过这些加工之后变换成上下文 DFD 的输出数据流。根据 0 层 DFD 中加工的复杂程度进一步建模加工的内容。

在建分层 DFD 时，根据需求情况可以将数据存储建模在不同层次的 DFD 中，注意要在绘制下层数据流图时要保持父图与子图平衡。父图中某加工的输入输出数据流必须与它的子图的输入输出数据流在数量和名字上相同，或者父图中的一个输入（或输出）数据流对应于子图中几个输入（或输出）数据流，而子图中组成这些数据流的数据项全体正好是父图中的这一条数据流。

【问题 1】

本题考查上下文 DFD，要求确定外部实体。在上下文 DFD 中，系统名称作为唯一加工的名称，外部实体和该唯一加工之间有输入输出数据流。通过考查系统的主要功能，不难发现，系统中涉及到客户和会议中心管理员，没有提到其他与系统交互的外部实体。根据描述（1）“客户提交预订请求后”，（2）“会议中心管理员收到客户预定请求的通知之后，提交确认”、“根据客户记录给客户发送临时预订确认信息和支付定金要求”等信息，对照图 1-1，从而即可确定 E1 为“客户”实体，E2 为“管理员”实体。

【问题 2】

本题要求确定图 1-2 所示的 0 层数据流图中的数据存储。重点分析说明中与数据存储有关的描述。根据（1）“客户提交预订请求后，检查预订表”，（2）“系统生成新临时

预订存入预订表，并对新客户创建一条客户信息记录加以保存”，可知 D1 为预订表、D2 为客户表；根据“会议中心提供举办会议的场地设施和各种设备”，（3）“根据临时预订或变更预定的设备和设施需求，分配所需设备（均能满足用户要求）和设施，更新相应的表和预订表”，“分配设施和设备”可知 D3 为和 D4 分别为场地（设施）表和设备表。

【问题 3】

本问题要求补充缺失的数据流及其起点和终点。

对照图 1-1 和图 1-2 的输入、输出数据流，数量不同，考查图 1-1 中从加工“会议预订系统”输出至 E1 的数据流，有“临时预订/预订/变更确认信息”，而图 1-2 中从加工输出至 E1 的数据流“临时预订确认信息”和“变更预订确认信息”，但缺少了其中一条数据流“预订确认信息”。

另外，图 1-1 中有“付款凭据”，图 1-2 中没有“付款凭据”，而只有“已支付定金凭据”，没有针对说明（7）中“管理员收到客户余款支付的通知后”中的“支付余款凭据”。上述两条数据流的遗失，使父图和子图数据流没有达到平衡。所以需要确定这两条条数据流或者其分解的数据流的起点或终点。

考查说明中的功能，先考查“确认预定”，功能（4）中“给客户发送预订确认信息”，对照图 1-2，加工 4 没有到实体 E1 客户的“预订确认信息”数据流；功能（7）中“管理员收到客户余款支付的通知后”，对照图 1-2，加工 7 没有从实体 E1 客户输入的数据流“余款支付凭据”。图中“余款支付凭据”数据流是上下文数据流图中数据流“支付凭据”的分解，与另一条分解出的数据流“已支付定金凭据”对照，改名为“已支付余款凭据”。

下面再仔细核对说明和图 1-2 之间是否还有遗失的数据流。

不难发现，功能（4）中“根据客户记录给客户发送预订确认信息”，而图 1-2 中加工 4 从 D1 预订表中读取预订信息，并没有读取客户信息，所以，此处遗失了数据流“客户记录”，起点是 D2 客户表，终点是加工 4 确认预订；功能（5）中“管理员确认变更后，根据客户记录给客户发送确认信息”，而图 1-2 中加工 5 并没有所根据的“客户记录”输入数据流，所以，此处遗失了数据流“客户记录”，起点是 D2 客户表，终点是加工 5 变更预订；功能（6）中“根据客户记录给满足条件的客户发送支付余款要求”，而图 1-2 中加工 6 并没有所根据的“客户记录”输入数据流，所以，此处遗失了数据流“客户记录”，起点是 D2 客户表，终点是加工 6 要求预订。

继续核对说明和图 1-2，不难发现，功能（6）中“管理员从预订表中查询距预订的会议时间两周内的预订”，而图 1-2 中没有从 D1 预订表到加工 6 的输入流，所以，此处遗失了数据流“距预订会议时间两周内的预订”，其起点是 D1 预订表，终点是加工 6 要求付款。

【问题 4】

DFD 中，外部实体可以是用户，也可以是与本系统交互的其他系统。如果某功能交

互的是外部系统（在本题中是 Email 系统），则本系统需要将发送给客户的确认信息发送给 Email 系统。然后由第三方 Email 系统向客户发送邮件，此时第三方 Email 系统即为外部实体，而非本系统内部加工，因此需要对图 1-1 和图 1-2 进行修改，添加外部实体“Email 系统”，并将数据流确认信息的终点全部改为 Email 系统。即将数据流“临时预订确认信息”“预订确认信息”“变更确认信息”数据流的终点改为新的外部实体“Email 系统”。

参考答案

【问题 1】

E1：客户　E2：管理员

【问题 2】

D1：预订表　D2：客户表

D3：场地表（设施表 或 场地设施表）

D4：设备表

注：D3 和 D4 可互换

【问题 3】

数　据　流	起　　点	终　　点
已支付余款凭据	E1 或 客户	7 或 支付余款
距预订会议时间两周内的预订	D1 或 预订表	6 或 要求付款
预订确认信息	4 或 确认预订	E1 或 客户
客户记录	D2 或 客户表	6 或 要求付款
客户记录	D2 或 客户表	5 或 变更预定
客户记录	D2 或 客户表	4 或 确认预定

注：上述 6 条数据流无顺序要求。

【问题 4】

将 Email 系统作为外部实体，并将发送给客户（E1）的确认信息数据流的终点全部改为 Email 系统（或具体说明确认信息数据流：临时预订确认信息、预订确认信息、变更确认信息，终点均改为 Email 系统）。

试题二（共 15 分）

阅读下列说明，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某销售公司当前的销售业务为商城实体店销售。现该公司拟开展网络销售业务，需要开发一个信息化管理系统。请根据公司现有业务及需求完成该系统的数据库设计。

【需求描述】

（1）记录公司所有员工的信息。员工信息包括工号、身份证号、姓名、性别、出生日期和电话，并只登记一部电话。

（2）记录所有商品的信息。商品信息包括商品名称、生产厂家、销售价格和商品介绍。系统内部用商品条码唯一区别每种商品。

（3）记录所有顾客的信息。顾客信息包括顾客姓名、身份证号、登录名、登录密码和电话号码。一位顾客只能提供一个电话号码。系统自动生成唯一的顾客编号。

（4）顾客登录系统之后，在网上商城购买商品。顾客可将选购的商品置入虚拟的购物车内，购物车可长期存放顾客选购的所有商品。顾客可在购物车内选择商品、修改商品数量后生成网购订单。订单生成后，由顾客选择系统提供的备选第三方支付平台进行电子支付，支付成功后系统需要记录唯一的支付凭证编号，然后由商城根据订单进行线下配送。

（5）所有的配送商品均由仓库统一出库。为方便顾客，允许每位顾客在系统中提供多组收货地址、收货人及联系电话。一份订单所含的多个商品可能由多名分检员根据商品所在仓库信息从仓库中进行分拣操作，分拣后的商品交由配送员根据配送单上的收货地址进行配送。

（6）新设计的系统要求记录实体店的每笔销售信息，包括营业员、顾客、所售商品及其数量。

【概念模型设计】

根据需求阶段收集的信息，设计的实体联系图（不完整）如图 2-1 所示。

【逻辑结构设计】

根据概念模型设计阶段完成的实体联系图，得出如下关系模式（不完整）：

员工（工号，身份证号，姓名，性别，出生日期，电话）

商品（商品条码，商品名称，生产厂家，销售价格，商品介绍，____(a)____）

顾客（顾客编号，姓名，身份证号，登录名，登录密码，电话）

收货地点（收货 ID，顾客编号，收货地址，收货人，联系电话）

购物车（顾客编号，商品条码，商品数量）

订单（订单 ID，顾客编号，商品条码，商品数量，____(b)____）

分检（分拣 ID，分拣员工号，____(c)____，分拣时间）

配送（配送 ID，分拣 ID，配送员工号，收货 ID，配送时间，签收时间，签收快照）

销售（销售 ID，营业员工号，顾客编号，商品条码，商品数量）

【问题 1】（4 分）

补充图 2-1 中的“配送”联系所关联的对象及联系类型。

【问题 2】（6 分）

补充逻辑结构设计中的（a）、（b）和（c）三处空缺。

【问题 3】（5 分）

对于实体店销售，若要增加送货上门服务，由营业员在系统中下订单，与网购的订单进行后续的统一管理。请根据该需求，对图 2-1 进行补充，并修改订单关系模式。

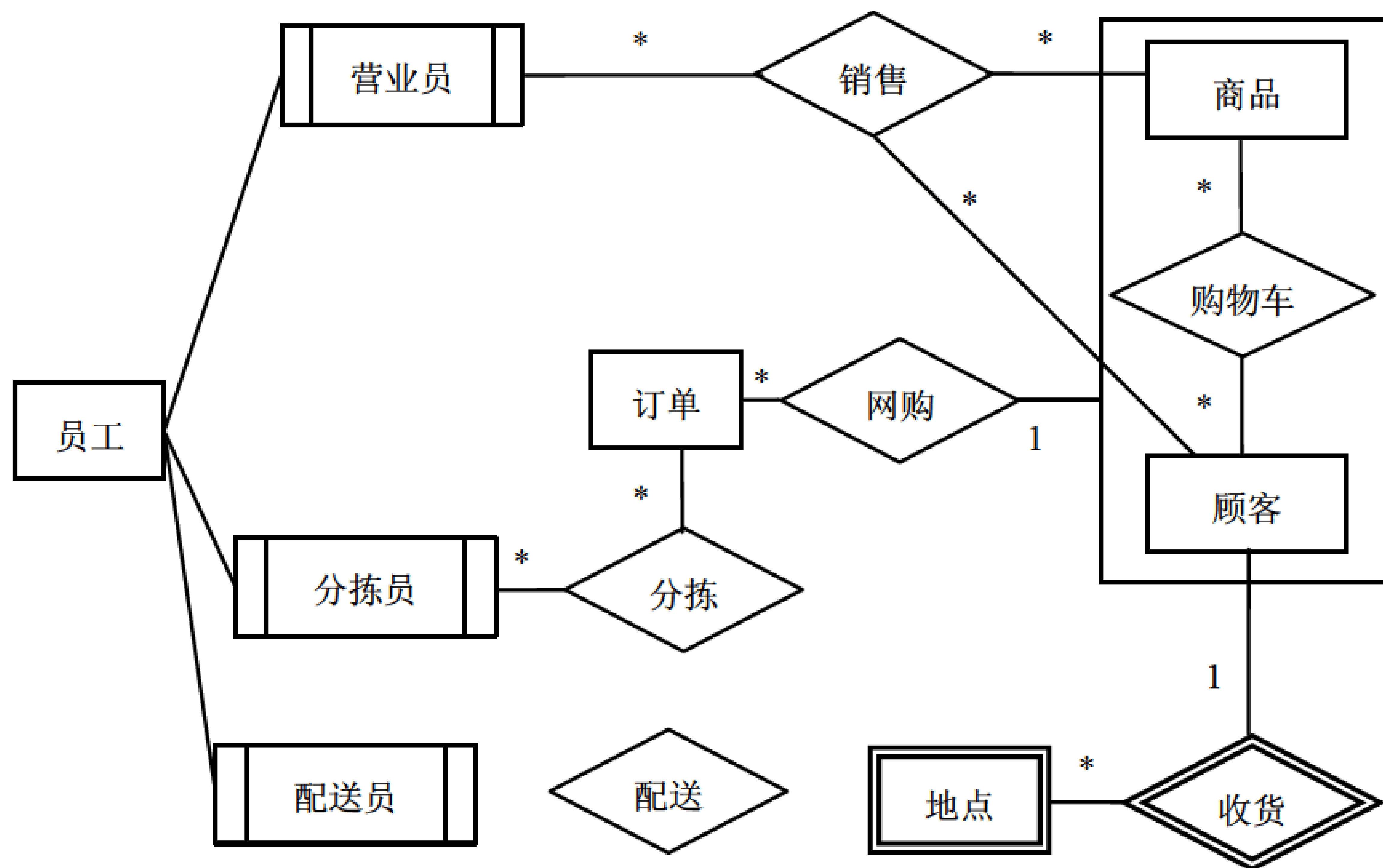


图 2-1　实体联系图

试题二分析

本题考查数据库概念结构设计和逻辑结构设计。

此类题目要求考生认真阅读题目中的需求描述，配合已给出的 E-R 图，理解概念结构设计中设计者对实体及联系的划分和组织方法，结合需求描述完成 E-R 图中空缺部分，并使用 E-R 图向关系模式的转换方法，完成逻辑结构设计。

【问题 1】

根据所给 E-R 图，结合需求描述，购物车作为顾客和商品之间的联系，而订单由顾客从购物车中选择商品生成，因此将购物车这一联系当作实体，与订单实体产生联系。将联系当作实体参与另一联系，称为聚合，通常当后一联系与此联系相关时，采用这种设计方法。顾客可以从购物车中生成多个订单，一个订单只能从一个购物车里提取商品，属于一对多联系。

根据需求描述中的“分拣后的商品交由配送员根据配送单上的收货地址进行配送。”可以知道，配送是与分拣联系相关的联系，同样的，将分拣联系进行聚合，参与配送联系，同时参与配送联系的还有配送员和地点，为多对多对多联系，语义为配送员根据分拣结果按照收货地点进行配送，与需求相符。

【问题 2】

本小题考核 E-R 图向关系模式的转换。由于 E-R 图中没有画出实体及联系的属性，需要根据需求描述进行补充。根据需求中的“一种商品只能放在一个仓库中”和“一份订单所含的多个商品可能由多名分拣员根据商品的所在仓库信息从仓库中进行分拣操

作”，可以确定“所在仓库”作为商品实体的属性，转入商品关系中。

订单关系由 E-R 图中的订单实体和一对多联系网购合并而成，取一方的主码，即购物车这一联系的主码，为参与该联系的实体的主码商品条码和顾客编号，加上网购联系的属性数量，并入到订单实体转成的关系模式中。订单 ID 为订单实体的标识符，订单实体的其他属性需要通过需求描述中获取。根据需求“订单生成后，由顾客选择系统提供的备选第三方支付平台进行电子支付，支付成功后系统需要记录唯一的支付凭证编号”，支付凭证编号应为订单的属性，转入订单关系中。

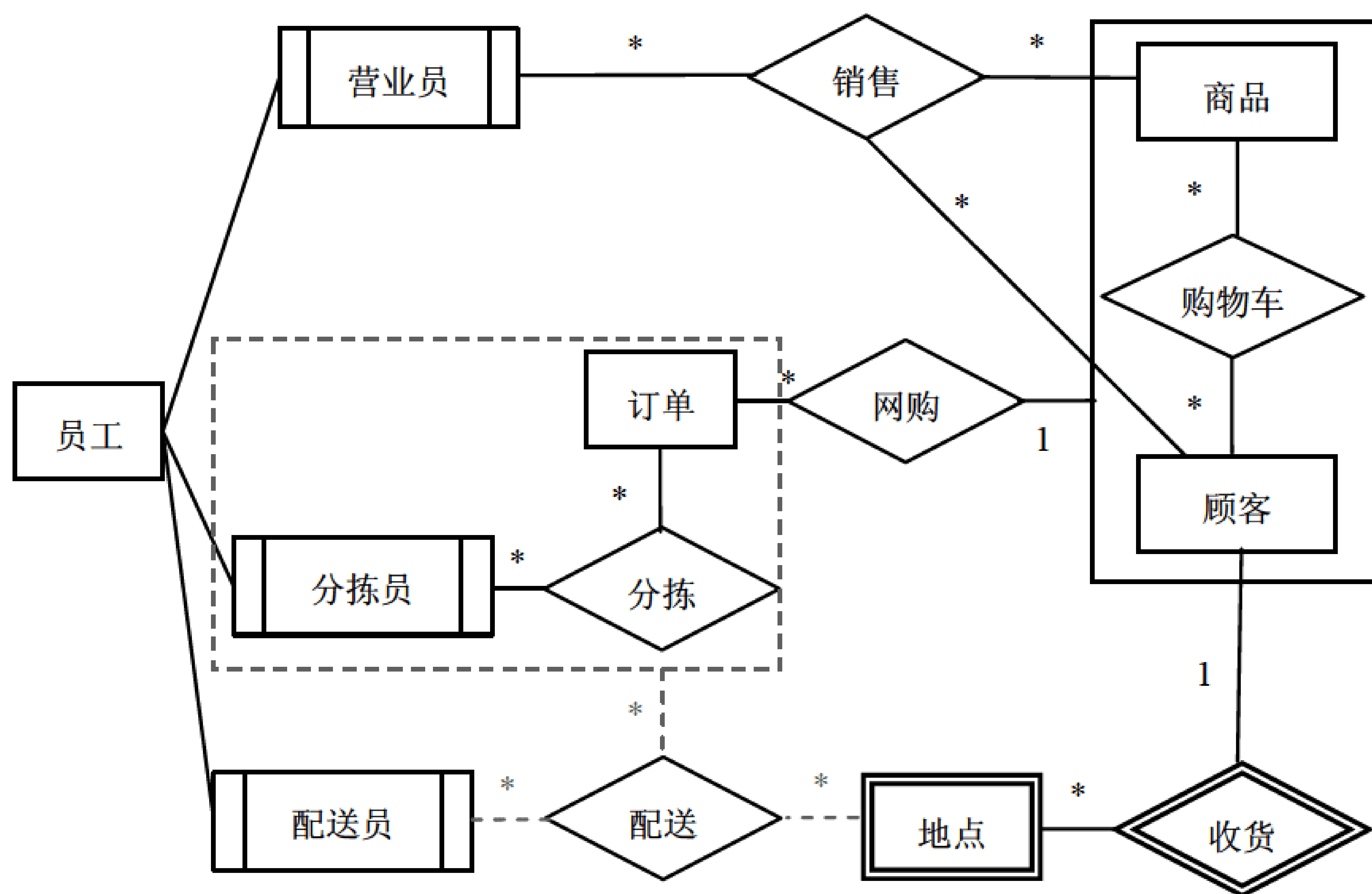
E-R 图中的分拣联系为分拣员与订单之间的多对多联系，转换成独立的分拣关系模式，应包含分拣员实体的标识符分拣员工号和订单实体的标识符订单 ID，及分拣联系的属性分拣时间。

【问题 3】

实体店的订单是营业员根据销售结果生成的，将销售联系聚合成实体，与订单产生联系。一笔销售对应一个订单，一个订单对应一笔销售，为一对一联系。转换为关系模式时，将此联系归入订单关系，即取销售的标识符销售 ID 加入到订单关系模式中。

参考答案

【问题 1】补充内容如图中虚线所示：

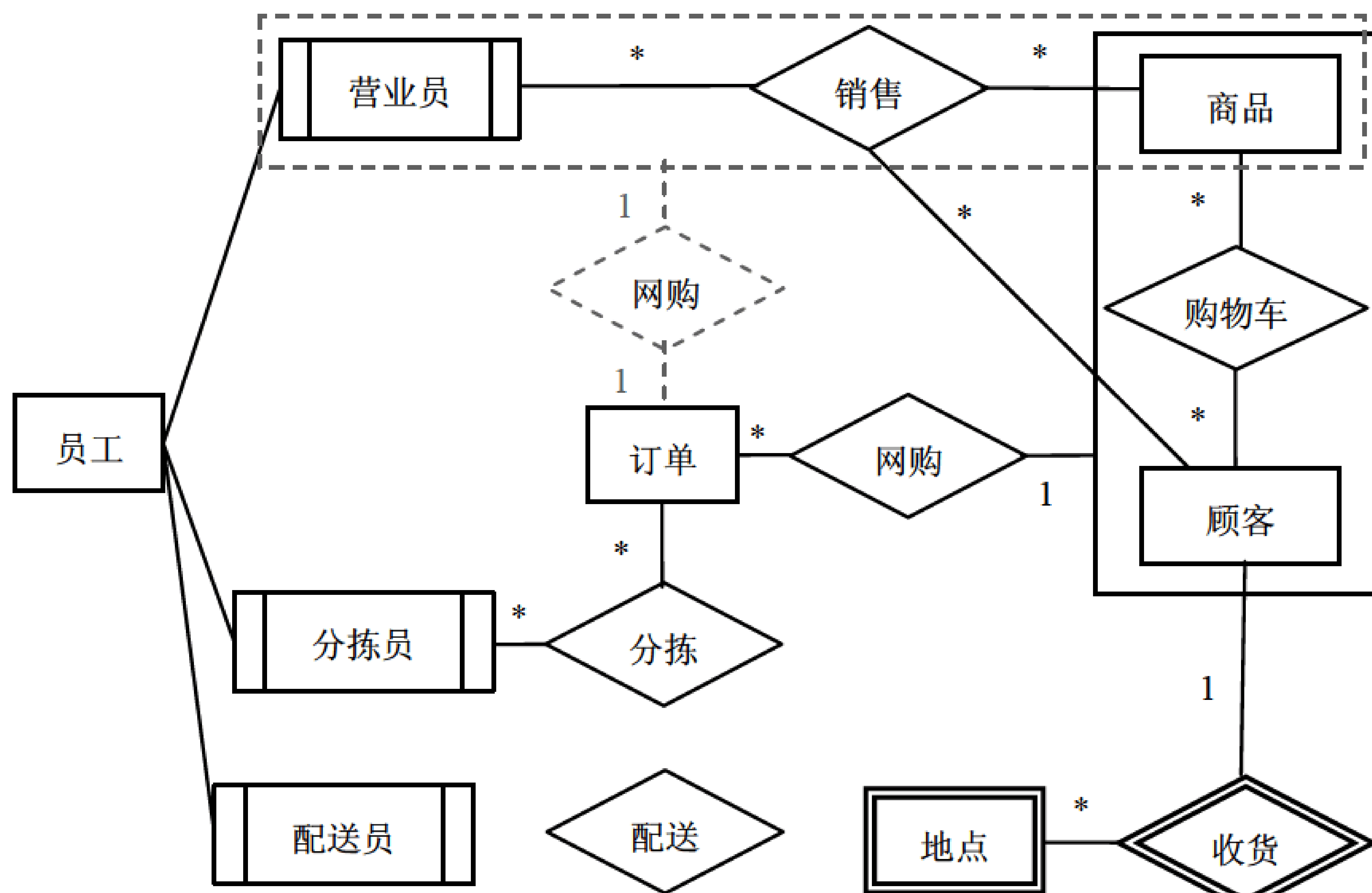


【问题 2】

（a）所在仓库　　（b）支付凭证　　（c）订单 ID

【问题 3】

补充内容如图中虚线所示：



关系模式：订单（订单 ID，顾客编号、商品条码，商品数量，销售 ID）

试题三（共 15 分）

阅读下列说明和图，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某软件公司欲设计实现一个虚拟世界仿真系统。系统中的虚拟世界用于模拟现实世界中的不同环境（由用户设置并创建），用户通过操作仿真系统中的 1~2 个机器人来探索虚拟世界。机器人维护着两个变量 b1 和 b2，用来保存从虚拟世界中读取的字符。

该系统的主要功能描述如下：

（1）机器人探索虚拟世界（Run Robots）。用户使用编辑器（Editor）编写文件以设置想要模拟的环境，将文件导入系统（Load File）从而在仿真系统中建立虚拟世界（Setup World）。机器人在虚拟世界中的行为也在文件中进行定义，建立机器人的探索行为程序（Setup Program）。机器人在虚拟世界中探索时（Run Program），有 2 种运行模式：

① 自动控制（Run）：事先编排好机器人的动作序列（指令（Instruction）），执行指令，使机器人可以连续动作。若干条指令构成机器人的指令集（Instruction Set）。

② 单步控制（Step）：自动控制方式的一种特殊形式，只执行指定指令中的一个动作。

（2）手动控制机器人（Manipulate Robots）。选定 1 个机器人后（Select Robot），可以采用手动方式控制它。手动控制有 4 种方式：

① Move：机器人朝着正前方移动一个交叉点。

② Left：机器人原地沿逆时针方向旋转 90 度。

③ Read：机器人读取其所在位置的字符，并将这个字符的值赋给 b1；如果这个位置上没有字符，则不改变 b1 的当前值。

④ Write：将 b1 中的字符写入机器人当前所在的位置，如果这个位置上已经有字符，该字符的值将会被 b1 的值替代。如果这时 b1 没有值，即在执行 Write 动作之前没有执行过任何 Read 动作，那么需要提示用户相应的错误信息（Show Errors）。

手动控制与单步控制的区别在于，单步控制时执行的是指令中的动作，只有一种控制方式，即执行下一个动作；而手动控制时有 4 种动作。

现采用面向对象方法设计并实现该仿真系统，得到如图 3-1 所示的用例图和图 3-2 所示的初始类图。图 3-2 中的类“Interpreter”和“Parser”用于解析描述虚拟世界的文件以及机器人行为文件中的指令集。

【问题 1】（6 分）

根据说明中的描述，给出图 3-1 中 U1～U6 所对应的用例名。

【问题 2】（4 分）

图 3-1 中用例 U1～U6 分别与哪个（哪些）用例之间有关系，是何种关系？

【问题 3】（5 分）

根据说明中的描述，给出图 3-2 中 C1～C5 所对应的类名。

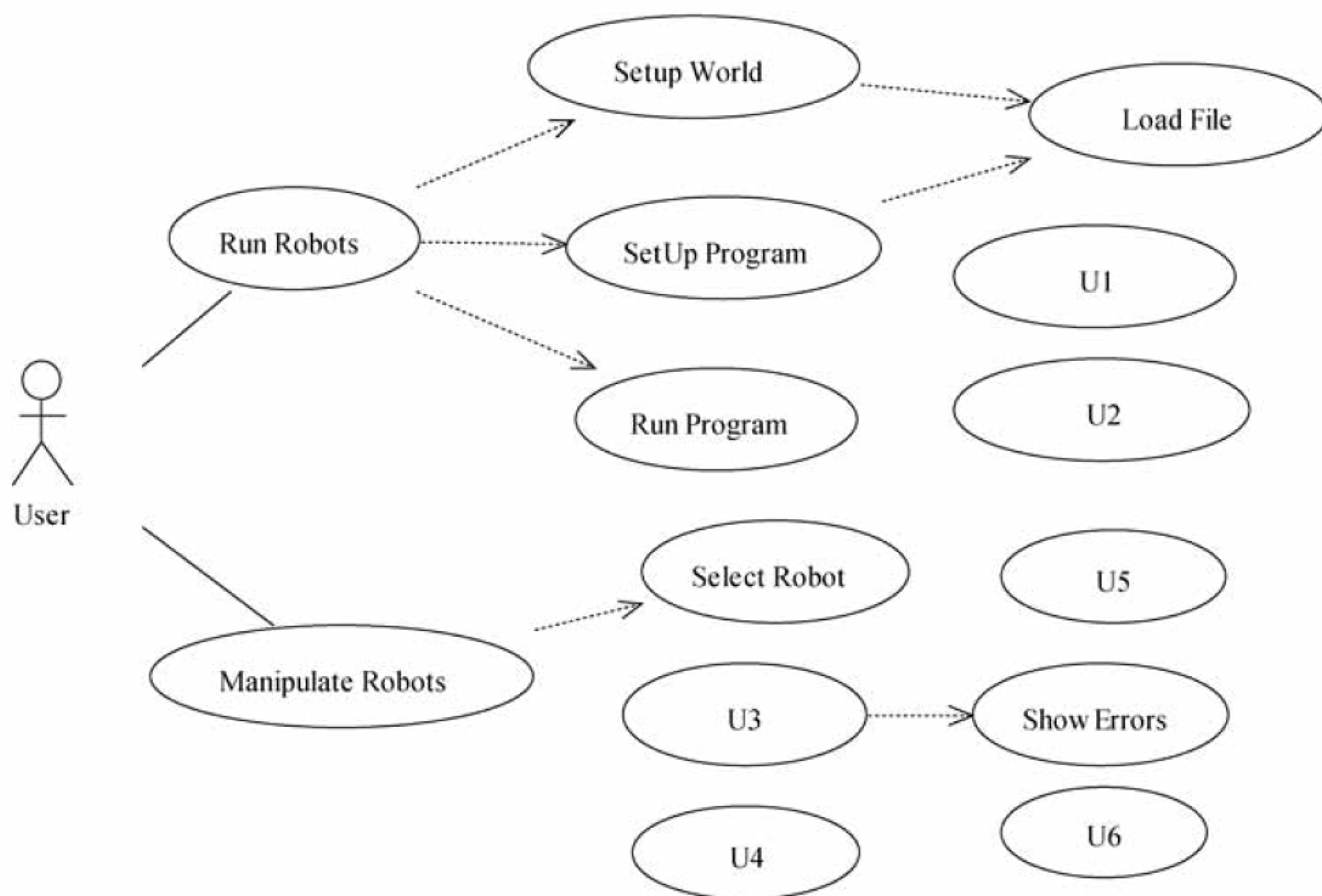


图 3-1　用例图

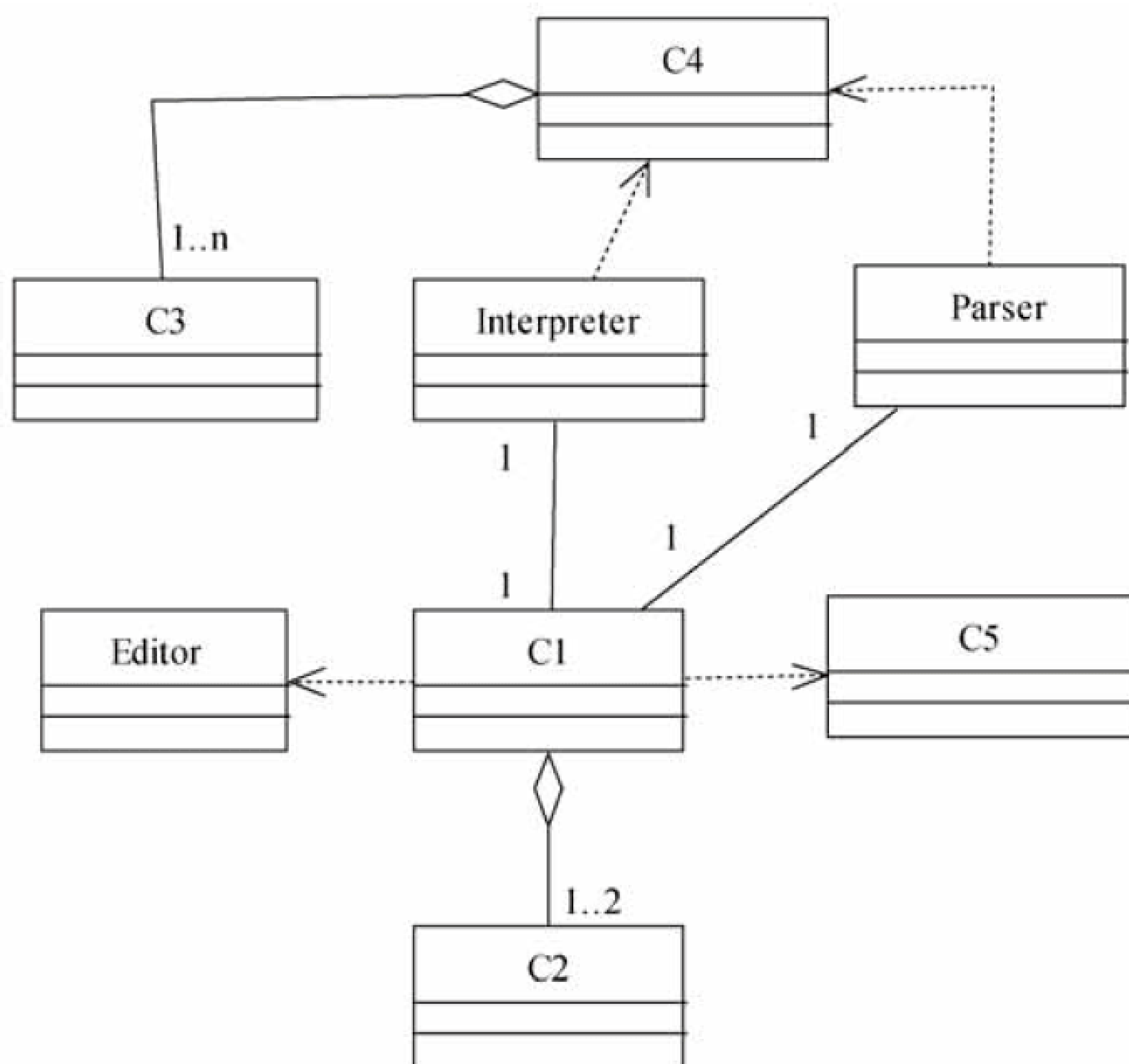


图 3-2　初始类图

试题三分析

本题属于经典的考题，主要考查面向对象分析方法与设计的基本概念。在建模方面，本题涉及到了 UML 的类图和用例图。本题的考点比较常规，题目难度不大。

【问题 1】

在图 3-1 的用例图中，需要确定 6 个用例。在面向对象方法中，用例及用例图是描述功能需求的工具，每一个用例表示一个单一的功能单元。通过对【说明】中功能描述的阅读，可以将未出现在图 3-1 中的功能单元列举出来：Run、Step、Move、Left、Read 和 Write。下面就是要判断这 6 个用例在图中的对应关系了。由图 3-1 可知，U3 中包含了 Show Errors 的功能，所以 U3 只能对应用例“Write”。其余的没有严格的顺序要求，但是在回答【问题 2】时要根据所填写的用例来判断用例之间的关系。这里我们按照下列顺序填写：U1->Run；U2->Step；U3->Write；U4->Read；U5->Left；U6->Move。

【问题 2】

图 3-1 中没有将用例之间的关系完整地给出来，因此需要根据【说明】中的功能描述判定 U1～U6 与其他用例之间的关系。根据【说明】中的描述可知，Run 和 Step 是 Run Program 的两种具体方式，所以这 3 个用例之间是有关系的，在 UML 用例图中，这种关联通常采用泛化关系描述。同理，U3～U6 用例是用例 Manipulate Robots 的 4 种具体实现方法，因此这 5 个用例之间也是泛化关系。

【问题 3】

本题要求将类图中缺失的 5 个类补充完整。在解答此类题目时，首先考虑类图中的特殊关系，如继承关系、聚集或组合关系等，这是比较好的突破口。另外应关注类之间的多重度。在图 3-2 中出现了两个聚集关系：C1 和 C2 之间以及 C3 和 C4 之间。我们先考虑 C1 和 C2 这一对，因为这两个类之间的多重度是一个具体的范围 1..2。【说明】中有一句话："用户通过操作仿真系统中的 1～2 个机器人来探索虚拟世界"，也就是说在虚拟世界中包含着 1-2 个机器人，由此可以推断 C2 对应的是机器人 Robot/ Robots，C1 代表的就是整个虚拟世界 World。

下面我们来看 C3 和 C4 这一对聚集关系。C4 和 Interpreter、Parser 有关联，而这两个类与文件及机器人指令集的解析有关，由此可以推断，C3、C4 这两个类也应该跟解析功能相关。由【说明】可知，系统中有两类需要解析的事物：虚拟世界文件和机器人指令集，而机器人指令集是由若干条指令构成的，这里就出现了一个聚集结构。因此 C3 应该对应 Instruction，C4 对应的是 InstructionSet。

对于最后一个类，将功能需求与用例图再回顾一遍，发现在类图中还缺少关于错误信息的描述，因此 C5 所对应的就是类 Error。

参考答案

【问题 1】

U1：Run　U2:　Step　U3: Write　U4: Read　U5: Left　U6: Move

注：U1 和 U2 可以互换；U4～U6 可以互换。

【问题 2】

U1～U2 与 Run Program 有关系；是泛化关系

U3～U6 与 Manipulate Robots 有关系；是泛化关系

【问题 3】

C1: World　C2: Robot/ Robots　C3: Instruction

C4: InstructionSet　C5: Error/Errors

试题四（共 15 分）

阅读下列说明和 C 代码，回答问题 1 至问题 3，将解答写在答题纸的对应栏内。

【说明】

在一块电路板的上下两端分别有 n 个接线柱。根据电路设计，用$(i,\pi(i))$表示将上端接线柱 i 与下端接线柱 $\pi(i)$相连，称其为该电路板上的第 i 条连线。如图 4-1 所示的 $\pi(i)$排列为{8, 7, 4, 2, 5, 1, 9, 3, 10, 6}。对于任何 $1\leqslant i<j\leqslant n$，第 i 条连线和第 j 条连线相交的充要条件是 $\pi(i)>\pi(j)$。

在制作电路板时，要求将这 n 条连线分布到若干绝缘层上，在同一层上的连线不相交。现在要确定将哪些连线安排在一层上，使得该层上有尽可能多的连线，即确定连线集 $Nets = \{(i,\pi(i)),\ 1\leqslant i\leqslant n\}$的最大不相交子集。

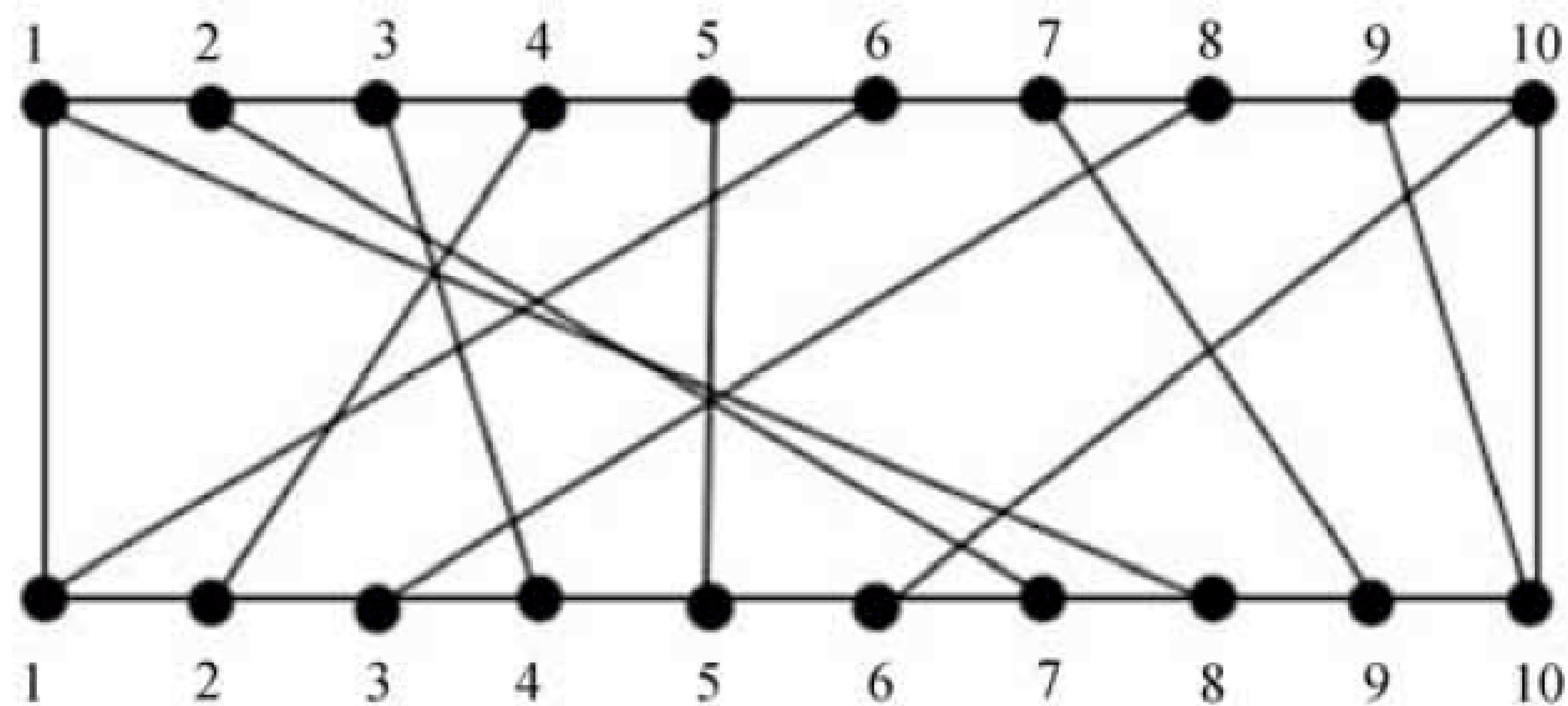


图 4-1 电路布线示意

【分析问题】

记 N(i, j) = {t | (t,π(t))∈Nets, t≤i,π(t)≤j}。N(i, j)的最大不相交子集为 MNS(i, j)，size(i, j) = |MNS(i, j)|。

经分析，该问题具有最优子结构性质。对规模为 n 的电路布线问题，可以构造如下递归式：

（1）当 i = 1 时，$size(1, j) = \begin{cases} 0 & j < \pi(1) \\ 1 & \text{其他情况} \end{cases}$

（2）当 i > 1 时，$size(i, j) = \begin{cases} size(i-1, j) & j < \pi(i) \\ \max\{size(i-1, j), size(i-1, \pi(i)-1)+1\} & \text{其他情况} \end{cases}$

【C 代码】

下面是算法的 C 语言实现。

（1）变量说明

size[i][j]：上下端分别有 i 个和 j 个接线柱的电路板的第一层最大不相交连接数

pi[i]：π(i)，下标从 1 开始

（2）C 程序

```
#include "stdlib.h"
#include <stdio.h>
#define   N   10          /* 问题规模 */
int m = 0;                /* 记录最大连接集合中的接线柱 */
void maxNum(int pi[], int size[N + 1][N + 1], int n)  {  /* 求最大不相交
连接数 */
    int i, j;
    for(j = 0; j < pi[1]; j++)    size[1][j] = 0;     /* 当 j < π(1)时  */
    for(j = pi[1]; j <= n; j++)  ____(1)____;         /* 当 j >= π(1)时 */
    for(i = 2; i < n; i++)  {
        for(j = 0; j < pi[i]; j++)  ____(2)____;      /* 当 j < pi[i]时 */
        for(j = pi[i]; j <= n; j++)  {  /* 当 j >= c[i]时，考虑两种情况 */
```

```
            size[i][j] = size[i-1][j] >= size[i-1][pi[i]-1]+1 ? size[i-1][j] :
            size[i-1][pi[i]-1]+1;
        }
    }
    /* 最大连接数 */
        size[n][n] = size[n-1][n] >= size[n-1][pi[n]-1]+1 ? size[n-1][n] :
        size[n-1][pi[n]-1]+1;
    }
    /* 构造最大不相交连接集合，net[i]表示最大不相交子集中第 i 条连线的上端接线柱的序号 */
    void constructSet(int pi[], int size[N + 1][N + 1], int n, int net[n]) {
        int i, j = n;
        m = 0;
        for(i = n; i > 1; i--)      {     /* 从后往前 */
            if(size[i][j] != size[i-1][j]) {  /* (i, pi[i])是最大不相交子集的一
            条连线 */
                ____(3)____;                  /*将 i 记录到数组 net 中，连接线数自增 1*/
                j = pi[i]-1;                  /* 更新扩展连线柱区间 */
            }
        }
        if(j >= pi[1])   net[m++] = 1;     /* 当 i =1 时 */
    }
```

【问题 1】（6 分）

根据以上说明和 C 代码，填充 C 代码中的空（1）～（3）。

【问题 2】（6 分）

根据题干说明和以上 C 代码，算法采用了＿（4）＿算法设计策略。

函数 maxNum 和 constructSet 的时间复杂度分别为＿（5）＿和＿（6）＿（用 O 表示）。

【问题 3】（3 分）

若连接排列为{8,7,4,2,5,1,9,3,10,6}，即如图 4-1 所示，则最大不相交连接数为＿（7）＿，包含的连线为＿（8）＿（用(i,π(i))的形式给出）。

试题四分析

本题考查算法设计和 C 语言实现算法的能力。

本题要求考试对常用的算法设计策略，包括分治法、动态规划、贪心算法、回溯法等有基本的掌握，并理解每类算法策略中的几个典型实例。

【问题 1】

一般不要求考生设计问题的求解算法，但要求考生能够理解题目给出的算法设计思路，并补充 C 程序。如本题中的空（1），可以根据题干中递归式第一部分和 C 代码中的注释，得到答案 size[1][j] = 1。

（1）当 i = 1 时，$size(1,j)=\begin{cases}0 & j<\pi(1)\\1 & \text{其他情况}\end{cases}$

空（2）则根据阅读题干中递归式第二部分和 C 代码中的注释，得到答案 size[i][j] = size[i-1][j]。

（2）当 i > 1 时，$size(i,j)=\begin{cases}size(i-1,j) & j<\pi(i)\\\max\{size(i-1,j),size(i-1,\pi(i)-1)+1\} & \text{其他情况}\end{cases}$

空（3）则依据 C 代码中的注释，即可得到答案 net[m++] = i。

【问题 2】

题干在叙述过程中，较明显的提到了动态规划策略的几个特点，如最优子结构，递归式，自底向上求解等，因此这是一个动态规划算法。算法的时间复杂度分析也较简单。函数 maxNum 中有两重循环，时间复杂度为 $O(n^2)$。函数 constructSet 中有一重循环，时间复杂度为 O(n)。

【问题 3】

本问题考查该算法的一个实例，理解了题干就可以直接计算出该实例的解，即最大不相交连接数为 4，连线为：(3,4)(5,5)(7,9)(9,10)。

参考答案

【问题 1】

（1）size[1][j] = 1

（2）size[i][j] = size[i-1][j]

（3）net[m++] = i　或其等价形式

【问题 2】

（4）动态规划

（5）$O(n^2)$

（6）O(n)

【问题 3】

（7）4

（8）(3,4)(5,5)(7,9)(9,10)

试题五（共 15 分）

阅读下列说明和 C++代码，将应填入＿（n）＿处的字句写在答题纸的对应栏内。

【说明】

某软件系统中，已设计并实现了用于显示地址信息的类 Address（如图 5-1 所示），现要求提供基于 Dutch 语言的地址信息显示接口。为了实现该要求并考虑到以后可能还会出现新的语言的接口，决定采用适配器（Adapter）模式实现该要求，得到如图 5-1 所示的类图。

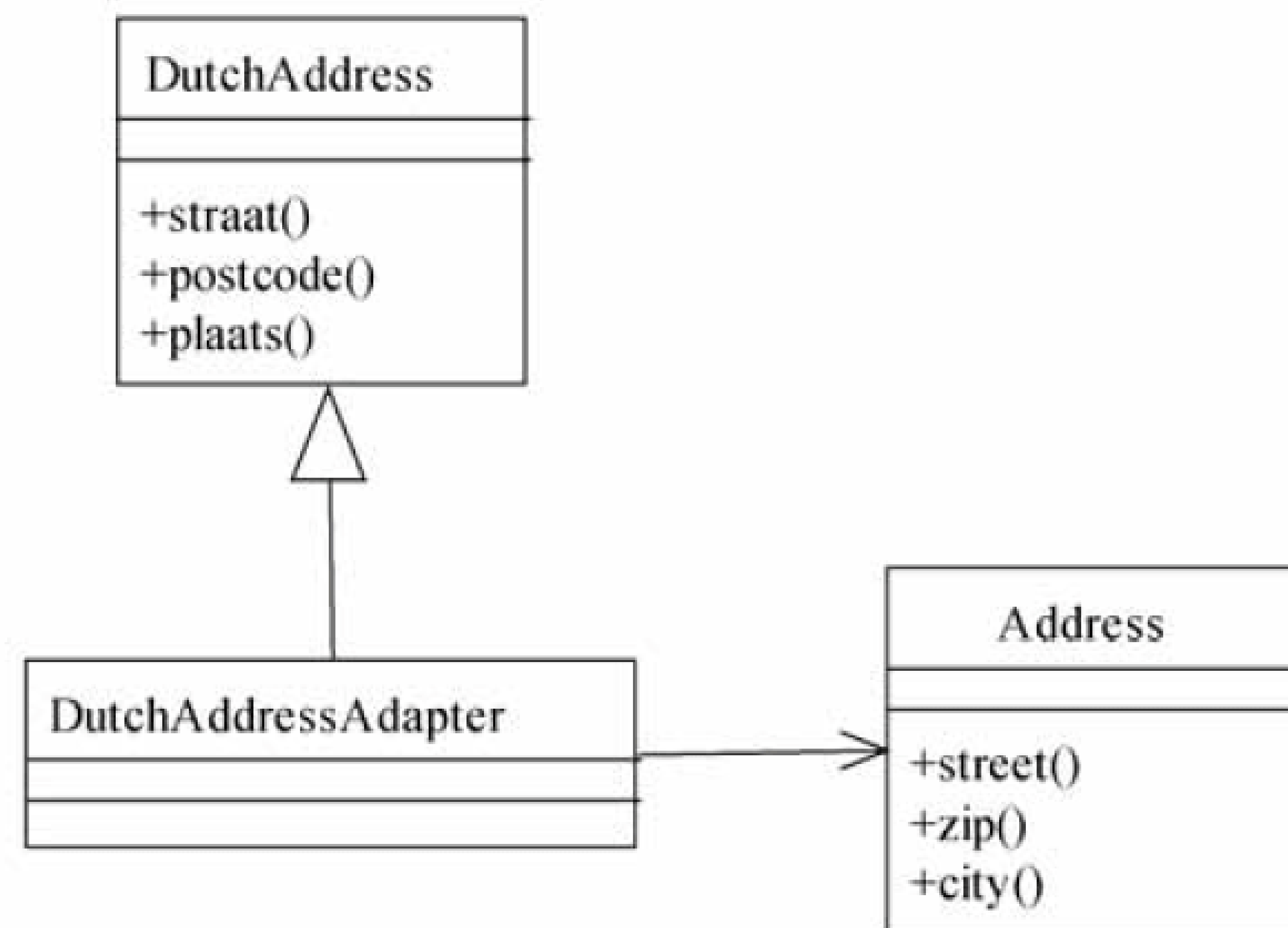


图 5-1　适配器模式类图

【C++代码】

```
#include <iostream>
using namespace std;

class Address {
public:
   void street()    {  /*  实现代码省略  */    }
   void zip()      {   /*  实现代码省略  */    }
   void city()     {   /*  实现代码省略  */    }
// 其他成员省略
};

class DutchAddress {
public:
   virtual void straat() = 0;
   virtual void postcode() = 0;
   virtual void plaats() = 0;
// 其他成员省略
};

class DutchAddressAdapter : public DutchAddress{
private:
   ____(1)____;
public:
    DutchAddressAdapter(Address *addr) {
        address = addr;
    }
    void straat() {
        ____(2)____;
```

```
        }
        void postcode() {
            ____(3)____;
        }
        void plaats() {
            ____(4)____;
        }
    // 其他成员省略
    };

    void testDutch(DutchAddress *addr) {
            addr->straat();
            addr->postcode();
            addr->plaats();
    }

    int main() {
        Address *addr = new Address();
        ____(5)____;
        cout << "\n The DutchAddress\n" << endl;
        testDutch(addrAdapter);
        return 0;
    }
```

试题五分析

本题考查Adapter（适配器）模式的基本概念和应用。

Adapter 模式的设计意图是，将一个类的接口转换成客户希望的另外一个接口。Adapter模式使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。

Adapter 模式有两种实现方式。类适配使用多重继承对一个接口与另一个接口进行匹配，其结构如图5-2所示。

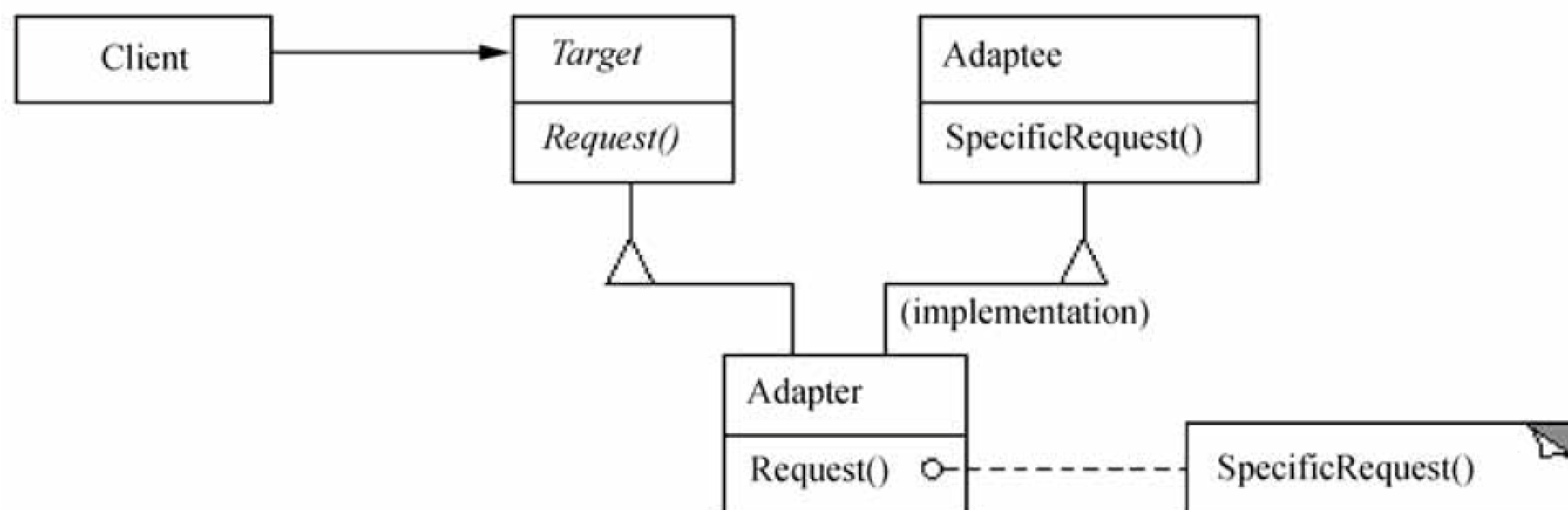


图5-2　类适配器结构图

对象适配器依赖于对象组合，其结构如图5-3所示。

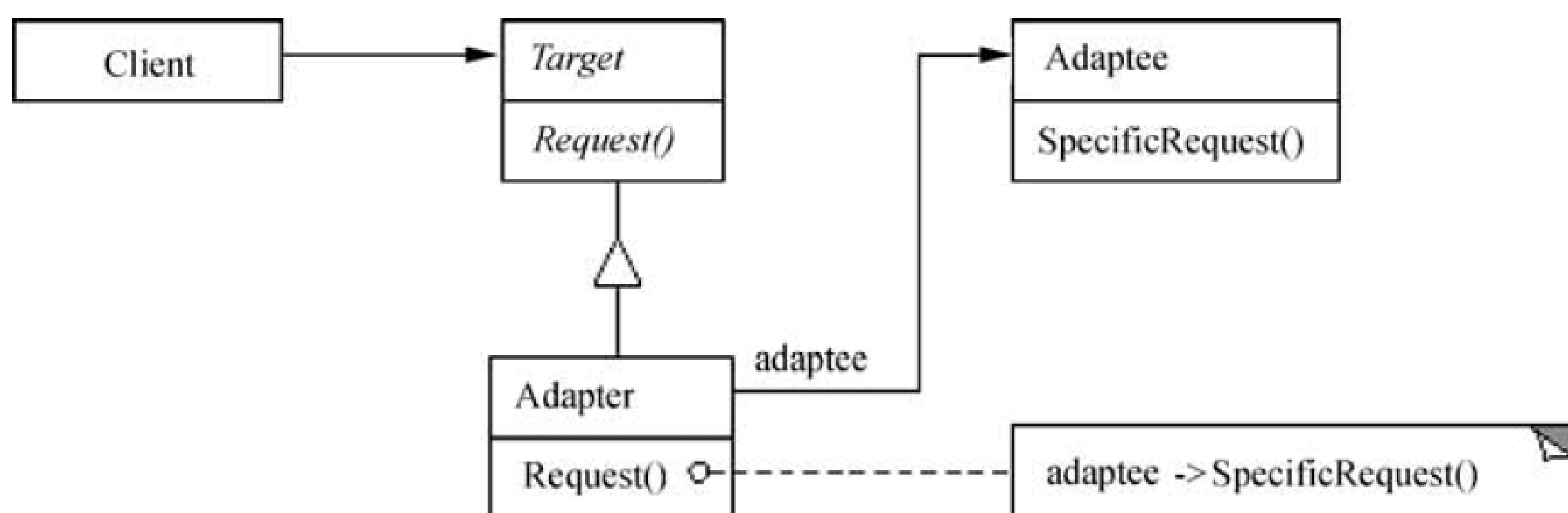


图 5-3　对象适配器结构图

- *Target* 定义 Client 使用的与特定领域相关的接口。
- Client 与符合 *Target* 接口的对象协同。
- Adaptee 定义一个已经存在的接口，这个接口需要适配。
- Adapter 对 Adaptee 的接口与 *Target* 接口进行适配。

Adapter 模式适用于：

- 想使用一个已经存在的类，而它的接口不符合要求。
- 想创建一个可以复用的类，该类可以与其他不相关的类或不可预见的类（即那些接口可能不一定兼容的类）协同工作。
- （仅适用于对象 Adapter）想使用一个已经存在的子类，但是不可能对每一个都进行子类化以匹配它们的接口。对象适配器可以适配它的父类接口。

本题中采用对象适配器，题中类 DutchAdddressAdapter 对应图 5-3 中的 Adapter、DutchAddress 对应图 5-3 中的 Target、Address 对应图 5-3 中的 Adaptee。

由图 5-3 可知，在 Adapter 中应该有一个 Adaptee 的对象，因此空（1）处应该填写的是 Address 的对象：Address *address。

类 DutchAddress 的实现采用了 C++中的抽象类，作为其子类 DutchAddressAdapter，必须对 DutchAddress 中的 3 个纯虚拟函数进行重置，所以空（2）～（4）是在考查这 3 个纯虚拟函数在子类中的实现方式。由图 5-3 可知，Adapter 中方法的实现方式还是要借助于 Adaptee 中所提供的行为，也就是说，DutchAddressAdapter 中 3 个纯虚拟函数的实现与 Address 是密不可分的。由此可知，空（2）～（4）分别应填入：address->street()、address->zip()和 address->city()。

第（5）空考查 Adapter 模式的使用。这里调用普通函数 testDutch 来进行测试，这个函数要求传递 DutchAddress 类型的参数，并且给出了实参的名字：addrAdatper。因此第（5）空应该填写的是 addrAdapter 的创建语句，这里需要使用到 DutchAddress 的构造函数。因此第（5）空应填写：DutchAddress *addrAdapter = new DutchAddressAdapter (addr)。

参考答案

（1）Address *address

（2）address->street()

（3）address->zip()

（4）address->city()

（5）DutchAddress *addrAdapter = new DutchAddressAdapter(addr)

试题六（共 15 分）

阅读下列说明和 Java 代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

【说明】

某软件系统中，已设计并实现了用于显示地址信息的类 Address（如图 6-1 所示），现要求提供基于 Dutch 语言的地址信息显示接口。为了实现该要求并考虑到以后可能还会出现新的语言的接口，决定采用适配器（Adapter）模式实现该要求，得到如图 6-1 所示的类图。

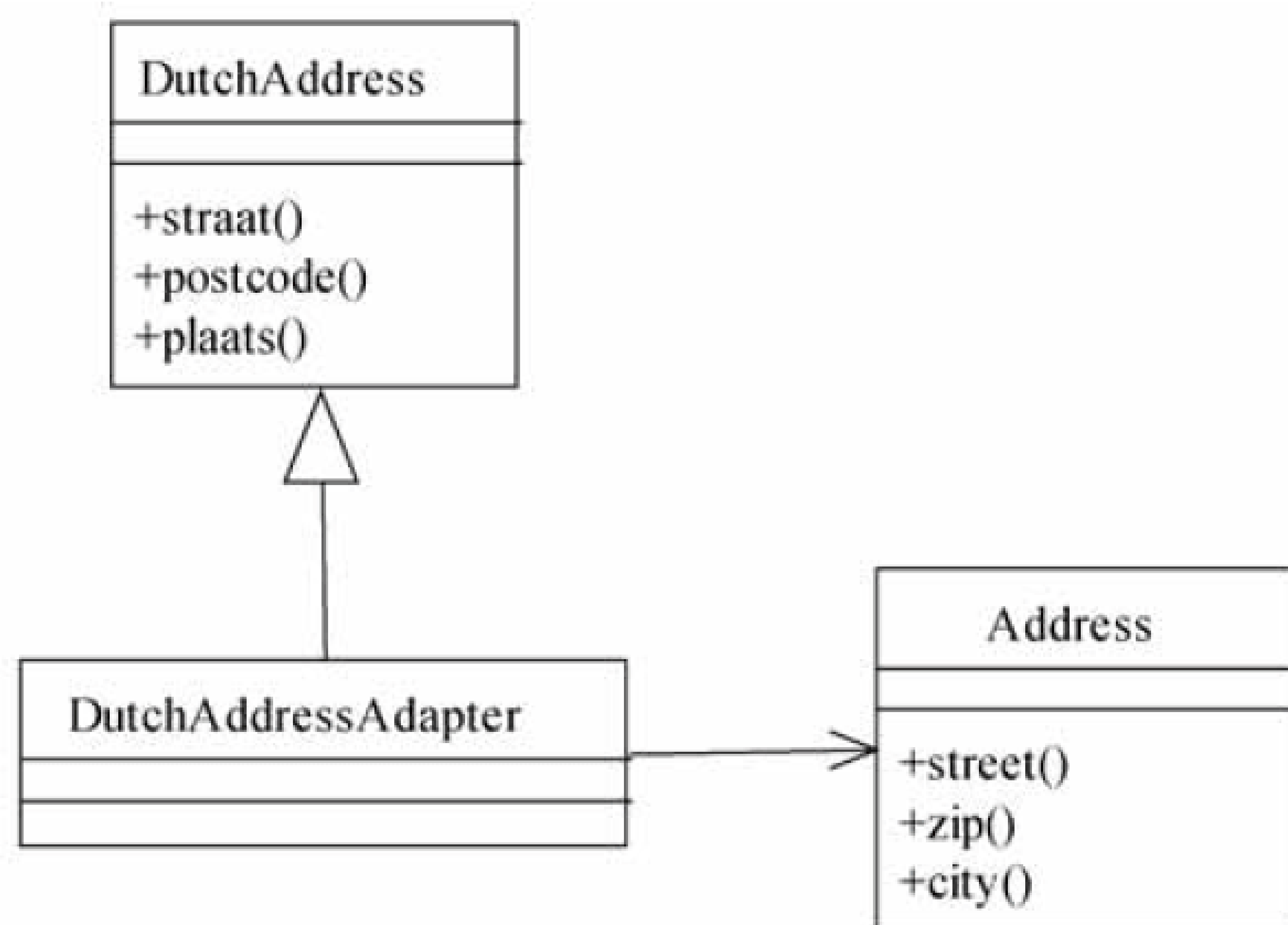


图 6-1　适配器模式类图

【Java 代码】

```
import java.util.*;

class Address {
    public void street()   {   //  实现代码省略        }
    public void zip()      {   //  实现代码省略        }
    public void city()     {   //  实现代码省略        }
// 其他成员省略
}

class DutchAddress {
```

```
    public void straat()    {  //  实现代码省略        }
    public void postcode()  {  //  实现代码省略        }
    public void plaats()    {  //  实现代码省略        }
// 其他成员省略
}

class DutchAddressAdapter extends DutchAddress {
    private ____(1)____;

    public DutchAddressAdapter(Address addr) {
        address = addr;
    }

    public void straat() {
        ____(2)____;
    }

    public void postcode() {
        ____(3)____;
    }

    public void plaats() {
        ____(4)____;
    }
// 其他成员省略
}

class Test {
    public static void main(String[] args) {
        Address addr = new Address();
        ____(5)____;
        System.out.println("\n The DutchAddress\n");
        testDutch(addrAdapter);
    }

    static void testDutch(DutchAddress addr) {
        addr.straat();
        addr.postcode();
        addr.plaats();
    }
}
```

试题六分析

本题考查 Adapter（适配器）模式的基本概念和应用。

Adapter 模式的设计意图是，将一个类的接口转换成客户希望的另外一个接口。Adapter 模式使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。

Adapter 模式有两种实现方式。类适配使用多重继承对一个接口与另一个接口进行匹配，其结构如图 6-2 所示。

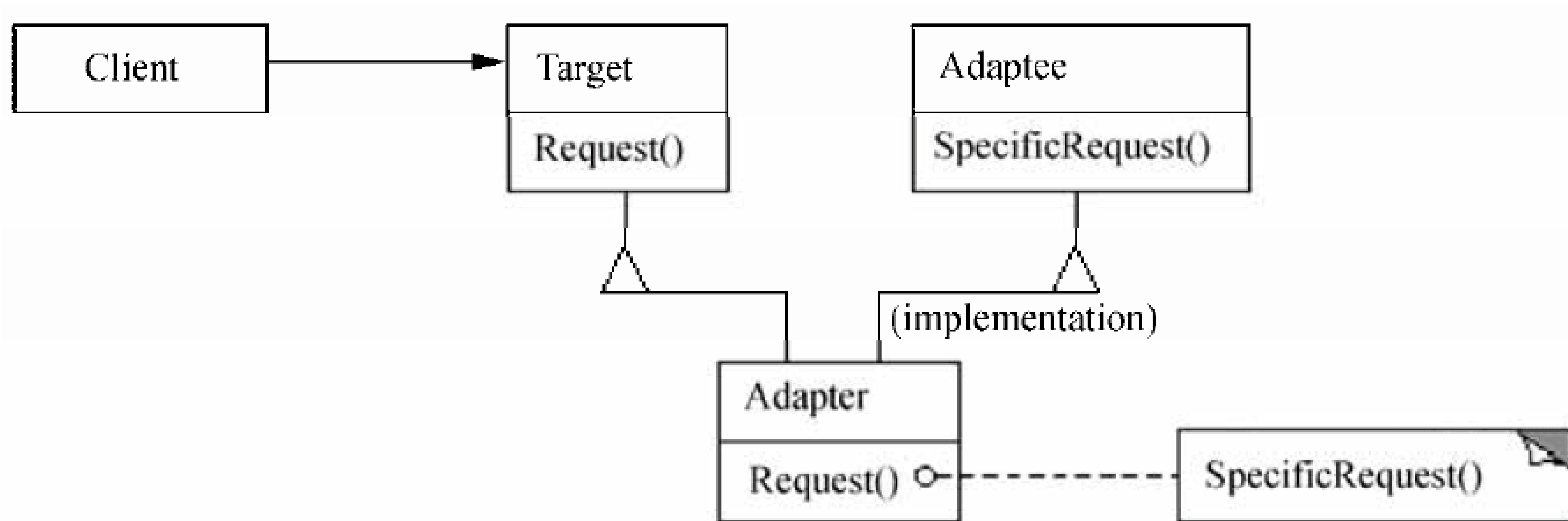


图 6-2 类适配器结构图

对象适配器依赖于对象组合，其结构如图 6-3 所示。

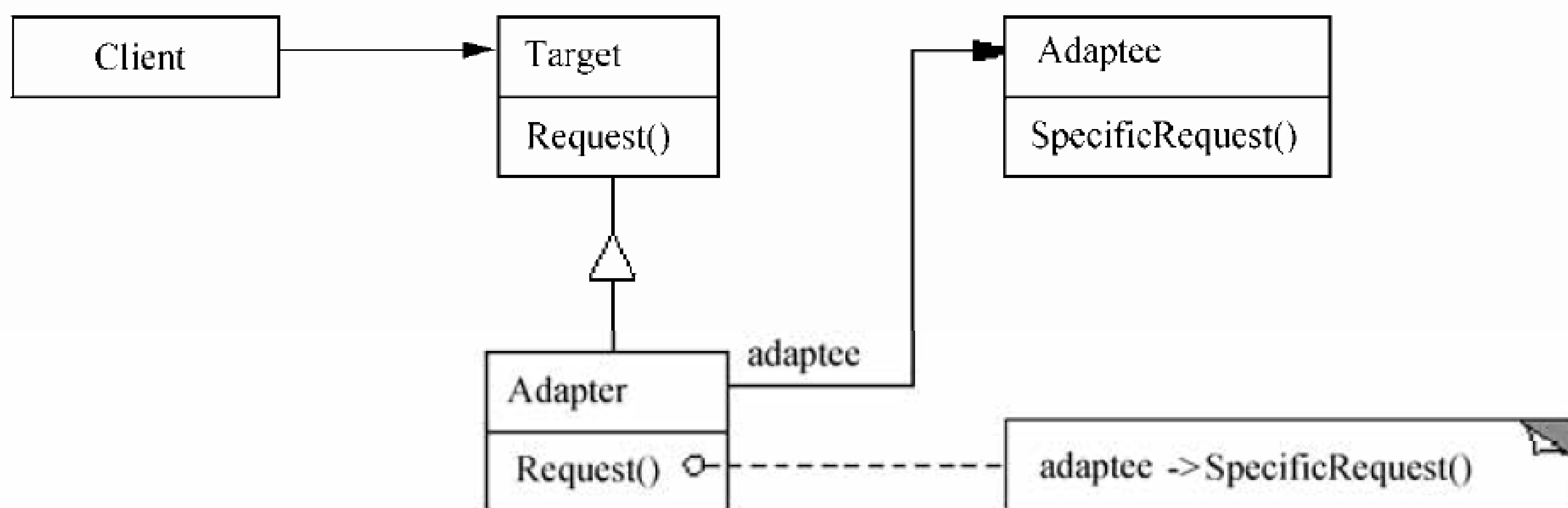


图 6-3 对象适配器结构图

- Target 定义 Client 使用的与特定领域相关的接口。
- Client 与符合 Target 接口的对象协同。
- Adaptee 定义一个已经存在的接口，这个接口需要适配。
- Adapter 对 Adaptee 的接口与 Target 接口进行适配。

Adapter 模式适用于：

- 想使用一个已经存在的类，而它的接口不符合要求。
- 想创建一个可以复用的类，该类可以与其他不相关的类或不可预见的类（即那些接口可能不一定兼容的类）协同工作。
- （仅适用于对象 Adapter）想使用一个已经存在的子类，但是不可能对每一个都进行子类化以匹配它们的接口。对象适配器可以适配它的父类接口。

本题中采用对象适配器，题中类 DutchAdddressAdapter 对应图 6-3 中的 Adapter、DutchAddress 对应图 6-3 中的 Target、Address 对应图 6-3 中的 Adaptee。

由图 6-3 可知，在 Adapter 中应该有一个 Adaptee 的对象，因此空（1）处应该填写的是 Address 的对象：Address address。

空（2）～（4）考查父类中的 3 个方法在子类 DutchAddressAdapter 的实现方式。由图 6-3 可知，Adapter 中方法的实现方式还是要借助于 Adaptee 中所提供的行为，也就是说，DutchAddressAdapter 中 3 个方法的实现与 Address 是密不可分的。由此可知，空（2）～（4）分别应填入：address.street()、address.zip()和 address.city()。

第（5）空考查 Adapter 模式的使用。这里使用方法 testDutch 来进行测试，这个方法要求传递 DutchAddress 类型的参数，并且给出了实参的名字：addrAdatper。因此第（5）空应该填写的是 addrAdapter 的创建语句，这里需要使用到 DutchAddress 的构造函数。因此第（5）空应填写：DutchAddress addrAdapter = new DutchAddressAdapter(addr)。

参考答案

（1）Address address

（2）address.street()

（3）address.zip()

（4）address.city()

（5）DutchAddress addrAdapter = new DutchAddressAdapter(addr)

第 31 章　2016 下半年软件设计师上午试题分析与解答

试题（1）

在程序运行过程中，CPU 需要将指令从内存中取出并加以分析和执行。CPU 依据 （1） 来区分在内存中以二进制编码形式存放的指令和数据。

（1）A．指令周期的不同阶段　　B．指令和数据的寻址方式

C．指令操作码的译码结果　　D．指令和数据所在的存储单元

试题（1）分析

本题考查计算机系统基础知识。

指令周期是执行一条指令所需要的时间，一般由若干个机器周期组成，是从取指令、分析指令到执行完所需的全部时间。CPU 执行指令的过程中，根据时序部件发出的时钟信号按部就班进行操作。在取指令阶段读取到的是指令，在分析指令和执行指令时，需要操作数时再去读操作数。

参考答案

（1）A

试题（2）

计算机在一个指令周期的过程中，为从内存读取指令操作码，首先要将 （2） 的内容送到地址总线上。

（2）A．指令寄存器（IR）　　B．通用寄存器（GR）

C．程序计数器（PC）　　D．状态寄存器（PSW）

试题（2）分析

本题考查计算机系统基础知识。

CPU 首先从程序计数器（PC）获得需要执行的指令地址，从内存（或高速缓存）读取到的指令则暂存在指令寄存器（IR），然后进行分析和执行。

参考答案

（2）C

试题（3）

设 16 位浮点数，其中阶符 1 位、阶码值 6 位、数符 1 位、尾数 8 位。若阶码用移码表示，尾数用补码表示，则该浮点数所能表示的数值范围是 （3） 。

（3）A．$-2^{64} \sim (1-2^{-8})2^{64}$　　B．$-2^{63} \sim (1-2^{-8})2^{63}$

C．$-(1-2^{-8})2^{64} \sim (1-2^{-8})2^{64}$　　D．$-(1-2^{-8})2^{63} \sim (1-2^{-8})2^{63}$

试题（3）分析

本题考查计算机系统基础知识。

浮点格式表示一个二进制数 N 的形式为 $N=2^E\times F$，其中 E 称为阶码，F 叫做尾数。在浮点表示法中，阶码通常为含符号的纯整数，尾数为含符号的纯小数。

指数为纯整数，阶符 1 位、阶码 6 位在补码表示方式下可表示的最大数为 63（2^6-1），最小数为−64（-2^6）。尾数用补码表示时最小数为−1、最大数为 $1-2^{-8}$，因此该浮点表示的最小数为 -2^{63}，最大数为 $(1-2^{-8})\times 2^{63}$。

参考答案

（3）B

试题（4）

已知数据信息为 16 位，最少应附加<u>（4）</u>位校验位，以实现海明码纠错。

（4）A．3　　B．4　　C．5　　D．6

试题（4）分析

本题考查计算机系统基础知识。

海明码是利用奇偶性来检错和纠错的校验方法。海明码的构成方法是：在数据位之间插入 k 个校验位，通过扩大码距来实现检错和纠错。

设数据位是 n 位，校验位是 k 位，则 n 和 k 必须满足以下关系：$2^k-1\geqslant n+k$

若数据信息为 n=16 位，则 k=5 是满足 $2^k-1\geqslant n+k$ 的最小值。

参考答案

（4）C

试题（5）

将一条指令的执行过程分解为取指、分析和执行三步，按照流水方式执行，若取指时间 $t_{取指}=4\Delta t$、分析时间 $t_{分析}=2\Delta t$、执行时间 $t_{执行}=3\Delta t$，则执行完 100 条指令，需要的时间为<u>（5）</u>Δt。

（5）A．200　　B．300　　C．400　　D．405

试题（5）分析

本题考查计算机系统基础知识。

对于该指令流水线，建立时间为 4Δt+2Δt+3Δt=9Δt，此后每 4Δt 执行完一条指令，即执行完 100 条指令的时间为 9Δt+99*4Δt=405Δt。

参考答案

（5）D

试题（6）

以下关于 Cache 与主存间地址映射的叙述中，正确的是<u>（6）</u>。

（6）A．操作系统负责管理 Cache 与主存之间的地址映射

B．程序员需要通过编程来处理 Cache 与主存之间的地址映射

C．应用软件对Cache与主存之间的地址映射进行调度

D．由硬件自动完成Cache与主存之间的地址映射

试题（6）分析

本题考查计算机系统基础知识。

存储系统采用Cache技术的主要目的是提高存储器的访问速度，因此是由硬件自动完成Cache与主存之间的地址映射。

参考答案

（6）D

试题（7）

可用于数字签名的算法是___（7）___。

（7）A．RSA　　B．IDEA　　C．RC4　　D．MD5

试题（7）分析

本题考查网络安全相关基础知识。

RSA基于大数定律，通常用于对消息摘要进行签名；IDEA和RC4适宜于进行数据传输加密；MD5为摘要算法。

参考答案

（7）A

试题（8）

___（8）___不是数字签名的作用。

（8）A．接收者可验证消息来源的真实性

B．发送者无法否认发送过该消息

C．接收者无法伪造或篡改消息

D．可验证接收者合法性

试题（8）分析

本题考查数字签名方面的基础知识。

数字签名用于通信的A、B双方，使得A向B发送签名的消息P，提供以下服务：

①B可以验证消息P确实是来源于A；

②A不能否认发送过消息P；

③B不能编造或改变消息P。

数字签名首先需要生成消息摘要，使用非对称加密算法以及私钥对摘要进行加密。接收方使用发送放的公钥对消息摘要进行验证。

参考答案

（8）D

试题（9）

在网络设计和实施过程中要采取多种安全措施，其中___（9）___是针对系统安全需求

的措施。

（9）A. 设备防雷击　　B. 入侵检测

C. 漏洞发现与补丁管理　　D. 流量控制

试题（9）分析

设备防雷击属于物理线路安全措施，入侵检测和流量控制属于网络安全措施，漏洞发现与补丁管理属于系统安全措施。

参考答案

（9）C

试题（10）

__（10）__的保护期限是可以延长的。

（10）A. 专利权　　B. 商标权　　C. 著作权　　D. 商业秘密权

试题（10）分析

发明专利权的期限为二十年，实用新型专利权和外观设计专利权的期限为十年，均自申请日起计算。专利保护的起始日是从授权日开始，有下列情形之一的，专利权在期限届满前终止：①没有按照规定缴纳年费的；②专利权人以书面声明放弃其专利权的。还有一种情况就是专利期限到期，专利终止时，保护自然结束。

商标权保护的期限是指商标专用权受法律保护的有效期限。我国注册商标的有效期为十年，自核准注册之日起计算。注册商标有效期满可以续展；商标权的续展是指通过一定程序，延续原注册商标的有效期限，使商标注册人继续保持其注册商标的专用权。

在著作权的期限内，作品受著作权法保护；著作权期限届满，著作权丧失，作品进入公有领域。

法律上对商业秘密的保密期限没有限制，只要商业秘密的四个基本特征没有消失，权利人可以将商业秘密一直保持下去。权利人也可以根据实际状况，为商业秘密规定适当的期限。

参考答案

（10）B

试题（11）

甲公司软件设计师完成了一项涉及计算机程序的发明。之后，乙公司软件设计师也完成了与甲公司软件设计师相同的涉及计算机程序的发明。甲、乙公司于同一天向专利局申请发明专利。此情形下，__（11）__是专利权申请人。

（11）A. 甲公司　　B. 甲、乙两公司

C. 乙公司　　D. 由甲、乙公司协商确定的公司

试题（11）分析

当两个以上的申请人分别就同样的发明创造申请专利的，专利权授给最先申请的人。如果两个以上申请人在同一日分别就同样的发明创造申请专利的，应当在收到专利

行政管理部门的通知后自行协商确定申请人。如果协商不成，专利局将驳回所有申请人的申请，即均不授予专利权。我国专利法规定："两个以上的申请人分别就同样的发明创造申请专利的，专利权授予最先申请的人。"我国专利法实施细则规定："同样的发明创造只能被授予一项专利。依照专利法第九条的规定，两个以上的申请人在同一日分别就同样的发明创造申请专利的，应当在收到国务院专利行政部门的通知后自行协商确定申请人。"

参考答案

（11）D

试题（12）

甲、乙两厂生产的产品类似，且产品都使用"B"商标。两厂于同一天向商标局申请商标注册，且申请注册前两厂均未使用"B"商标。此情形下，<u>（12）</u>能核准注册。

（12）A. 甲厂　　B. 由甲、乙厂抽签确定的厂
　　　C. 乙厂　　D. 甲、乙两厂

试题（12）分析

我国商标注册以申请在先为原则，使用在先为补充。当两个或两个以上申请人在同一种或者类似商品上申请注册相同或者近似商标时，申请在先的人可以获得注册。对于同日申请的情况，商标法及其实施条例规定保护先用人的利益，使用在先的人可以获得注册"使用"包括将商标用于商品、商品包装、容器以及商品交易书上，或者将商标用于广告宣传、展览及其他商业活动中。如果同日使用或均未使用，则采取申请人之间协商解决，不愿协商或者协商不成的，由各申请人抽签决定。商标局通知各申请人以抽签的方式确定一个申请人，驳回其他人的注册申请。商标局已经通知但申请人未参加抽签的，视为放弃申请。

参考答案

（12）B

试题（13）、（14）

在 FM 方式的数字音乐合成器中，改变数字载波频率可以改变乐音的<u>（13）</u>，改变它的信号幅度可以改变乐音的<u>（14）</u>。

（13）A. 音调　　B. 音色　　C. 音高　　D. 音质
（14）A. 音调　　B. 音域　　C. 音高　　D. 带宽

试题（13）、（14）分析

音调（Pitch）用来表示人的听觉分辨一个声音的调子高低的程度，主要由声音的频率决定，同时也与声音强度有关。对一定强度的纯音，音调随频率的升降而升降；对一定频率的纯音、低频纯音的音调随声强增加而下降，高频纯音的音调却随强度增加而上升。

音色（Timbre）是指声音的感觉特性，不同的人声和不同的声响都能区分为不同的

音色，即音频泛音或谐波成分。

音高是指各种不同高低的声音（即音的高度），是音的基本特征的一种。

在 FM 方式音乐合成器中，数字载波波形和调制波形有很多种，不同型号的 FM 合成器所选用的波形也不同。各种不同乐音的产生是通过组合各种波形和各种波形参数并采用各种不同的方法实现的。改变数字载波频率可以改变乐音的音调，改变它的幅度可以改变乐音的音高。

参考答案

（13）A　（14）C

试题（15）

结构化开发方法中，<u>（15）</u>主要包含对数据结构和算法的设计。

（15）A. 体系结构设计　B. 数据设计　C. 接口设计　D. 过程设计

试题（15）分析

本题考查软件设计的基础知识。

结构化设计主要包括：

① 体系结构设计：定义软件的主要结构元素及其关系。

② 数据设计：基于实体联系图确定软件涉及的文件系统的结构及数据库的表结构。

③ 接口设计：描述用户界面，软件和其他硬件设备、其他软件系统及使用人员的外部接口，以及各种构件之间的内部接口。

④ 过程设计：确定软件各个组成部分内的算法及内部数据结构，并选定某种过程的表达形式来描述各种算法。

参考答案

（15）D

试题（16）

在敏捷过程的开发方法中，<u>（16）</u>使用了迭代的方法，其中，把每段时间（30 天）一次的迭代称为一个“冲刺”，并按需求的优先级别来实现产品，多个自组织和自治的小组并行地递增实现产品。

（16）A. 极限编程 XP　B. 水晶法
C. 并列争球法　D. 自适应软件开发

试题（16）分析

本题考查敏捷方法的基础知识。

在 20 世纪 90 年代后期，一些开发人员抵制严格化软件开发过程，试图强调灵活性在快速有效的软件生产中的作用，提出了敏捷宣言，即个人和交互胜过过程和工具；可以运行的软件胜过面面俱到的文档；与客户合作胜过合同谈判；对变化的反应胜过遵循计划。

基于这些基本思想，有很多敏捷过程的典型方法。其中，极限编程 XP 是激发开发

人员创造性、使得管理负担最小的一组技术；水晶法 Crystal 认为每一个不同的项目都需要一套不同的策略、约定和方法论；并列争球法（Scrum）使用迭代的方法，其中把每 30 天一次的迭代成为一个冲刺，并按需求的优先级来实现产品。多个自组织和自治小组并行地递增实现产品，并通过简短的日常情况会议进行协调。

自适应软件开发（ASD）有六个基本的原则：

① 在自适应软件开发中，有一个使命作为指导，它设立了项目的目标，但并不描述如何达到这个目标；

② 特征被视为客户键值的关键，因此，项目是围绕着构造的构件来组织并实现特征；

③ 过程中的迭代是很重要的，因此重做与做同样重要，变化也包含其中；

④ 变化不视为是一种更正，而是对软件开发实际情况的调整；

⑤ 确定的交付时间迫使开发人员认证考虑每一个生产版本的关键需求；

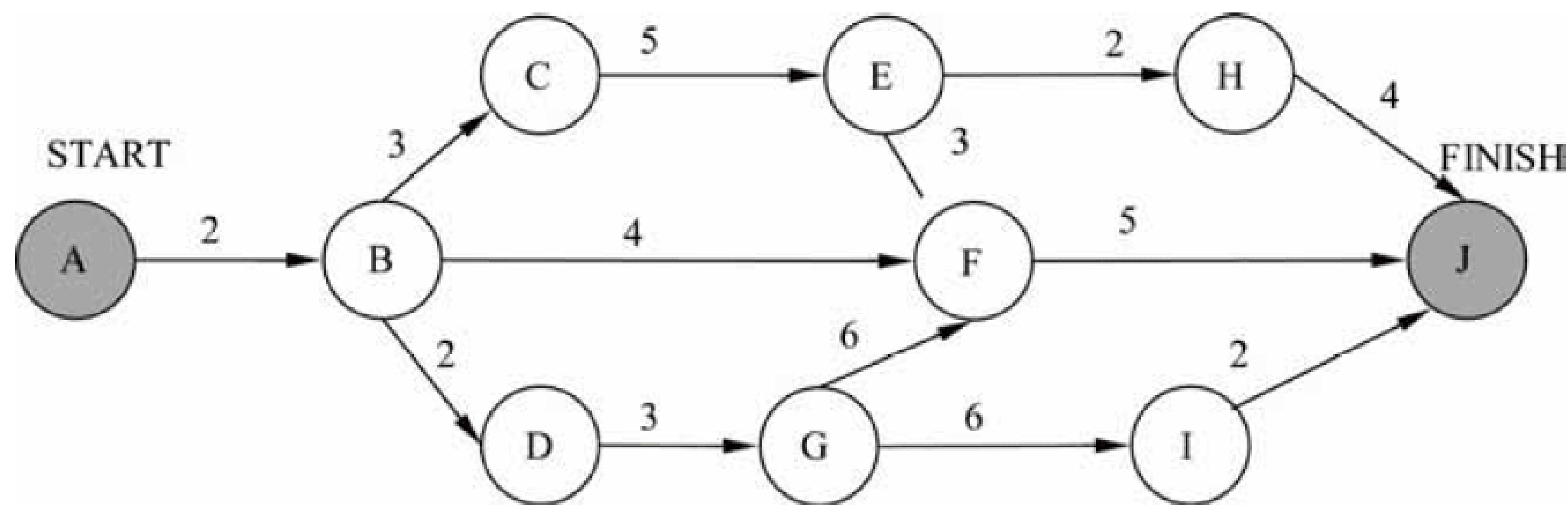
⑥ 风险也包含其中，它使开发人员首先跟踪最艰难的问题。

参考答案

（16）C

试题（17）、（18）

某软件项目的活动图如下图所示，其中顶点表示项目里程碑，连接顶点的边表示包含的活动，边上的数字表示相应活动的持续时间（天），则完成该项目的最少时间为__（17）__天。活动 BC 和 BF 最多可以晚开始__（18）__天而不会影响整个项目的进度。



（17）A．11　　B．15　　C．16　　D．18

（18）A．0 和 7　　B．0 和 11　　C．2 和 7　　D．2 和 11

试题（17）、（18）分析

本题考查软件项目管理的基础知识。

活动图是描述一个项目中各个工作任务相互依赖关系的一种模型，项目的很多重要特性可以通过分析活动图得到，如估算项目完成时间，计算关键路径和关键活动等。

根据上图计算出关键路径为 A-B-C-E-F-J 和 A-B-D-G-F-J，其长度为 18。关键路径上的活动均为关键活动。活动 BC 在关键路径上，因此松弛时间为 0。活动 BF 不在关键

路径上，包含该活动的最长路径为 A-B-F-J，其长度为 11，因此该活动的松弛时间为 18−11=7。

参考答案

（17）D　（18）A

试题（19）

成本估算时，__（19）__方法以规模作为成本的主要因素，考虑多个成本驱动因子。该方法包括三个阶段性模型，即应用组装模型、早期设计阶段模型和体系结构阶段模型。

（19）A．专家估算　B．Wolverton　C．COCOMO　D．COCOMO II

试题（19）分析

本题考查软件项目管理的基础知识。

存在多种软件项目管理的成本估算方法。其中专家估算方法主要依赖于专家的背景和经验，具有较大的主观性。Wolverton 模型基于一个成本矩阵，定义不同的软件类型（如控制、输入/输出等）和难易（容易和困难）的成本，基于此计算软件开发的成本。COCOMO 模型将规模视为成本的主要因素，考虑多个成本驱动因子。在后来的版本 COCOMO II 中，还考虑了软件开发的不同阶段，包含三个阶段性模型，即应用组装模型、早期设计阶段模型和体系结构阶段模型。

参考答案

（19）D

试题（20）

逻辑表达式求值时常采用短路计算方式。“&&”“||”“!”分别表示逻辑与、或、非运算，“&&”“||”为左结合，“!”为右结合，优先级从高到低为“!”“&&”“||”。对逻辑表达式“x && (y || ! z)”进行短路计算方式求值时，__（20）__。

（20）A．x 为真，则整个表达式的值即为真，不需要计算 y 和 z 的值

B．x 为假，则整个表达式的值即为假，不需要计算 y 和 z 的值

C．x 为真，再根据 z 的值决定是否需要计算 y 的值

D．x 为假，再根据 y 的值决定是否需要计算 z 的值

试题（20）分析

本题考查逻辑运算知识。

由“逻辑与”“逻辑或”运算构造的逻辑表达式可采用短路计算的方式求值。

“逻辑与”运算“&&”的短路运算逻辑为：a&&b 为真当且仅当 a 和 b 都为真，当 a 为假，无论 b 的值为真还是假，该表达式的值即为假，也就是说此时不需要再计算 b 的值。

“逻辑或”运算“||”的短路运算逻辑为：a||b 为假当且仅当 a 和 b 都为假，当 a 为真，无论 b 的值为真还是假，该表达式的值即为真，也就是说此时不需要再计算 b 的值。

对逻辑表达式“x && (y || ! z)”进行短路计算方式求值时，x 为假则整个表达式的值

即为假，不需要计算 y 和 z 的值。若 x 的值为真，则再根据 y 的值决定是否需要计算 z 的值，y 为真就不需要计算 z 的值，y 为假则需要计算 z 的值。

参考答案

（20）B

试题（21）

常用的函数参数传递方式有传值与传引用两种。__（21）__。

（21）A．在传值方式下，形参与实参之间互相传值

B．在传值方式下，实参不能是变量

C．在传引用方式下，修改形参实质上改变了实参的值

D．在传引用方式下，实参可以是任意的变量和表达式

试题（21）分析

本题考查程序语言基础知识。

传值调用和引用调用是常用的两种参数传递方式。在传值调用方式下，是将实参的值传递给形参，该传递是单方向的，调用结束后不会再将形参的值传给实参。在引用调用方式下，实质上是将实参的地址传递给形参，借助指针在间接访问数据方式下（或者将形参看作是实参的别名），在被调用函数中对形参的修改实质上是对实参的修改。

参考答案

（21）C

试题（22）

二维数组 a[1..N, 1..N]可以按行存储或按列存储。对于数组元素 a[i,j]（1≤i,j≤N），当__（22）__时，在按行和按列两种存储方式下，其偏移量相同。

（22）A．i≠j B．i=j C．i>j D．i<j

试题（22）分析

本题考查数据存储知识。

二维数组 a[1..N, 1..N]用来表示一个 N*N 的方阵，主对角线上元素的行下标和列下标相同，以 4*4 的矩阵为例，如下图所示。

a_{11}	a_{12}	a_{13}	a_{14}
a_{21}	a_{22}	a_{23}	a_{24}
a_{31}	a_{32}	a_{33}	a_{34}
a_{41}	a_{42}	a_{43}	a_{44}

对于主对角线中的元素，无论按行方式排列还是按列方式排列，其在序列中的位置都是相同的。

参考答案

（22）B

试题（23）

实时操作系统主要用于有实时要求的过程控制等领域。实时系统对于来自外部的事件必须在__（23）__。

（23）A. 一个时间片内进行处理
B. 一个周转时间内进行处理
C. 一个机器周期内进行处理
D. 被控对象规定的时间内做出及时响应并对其进行处理

试题（23）分析

本题考查操作系统基础知识。

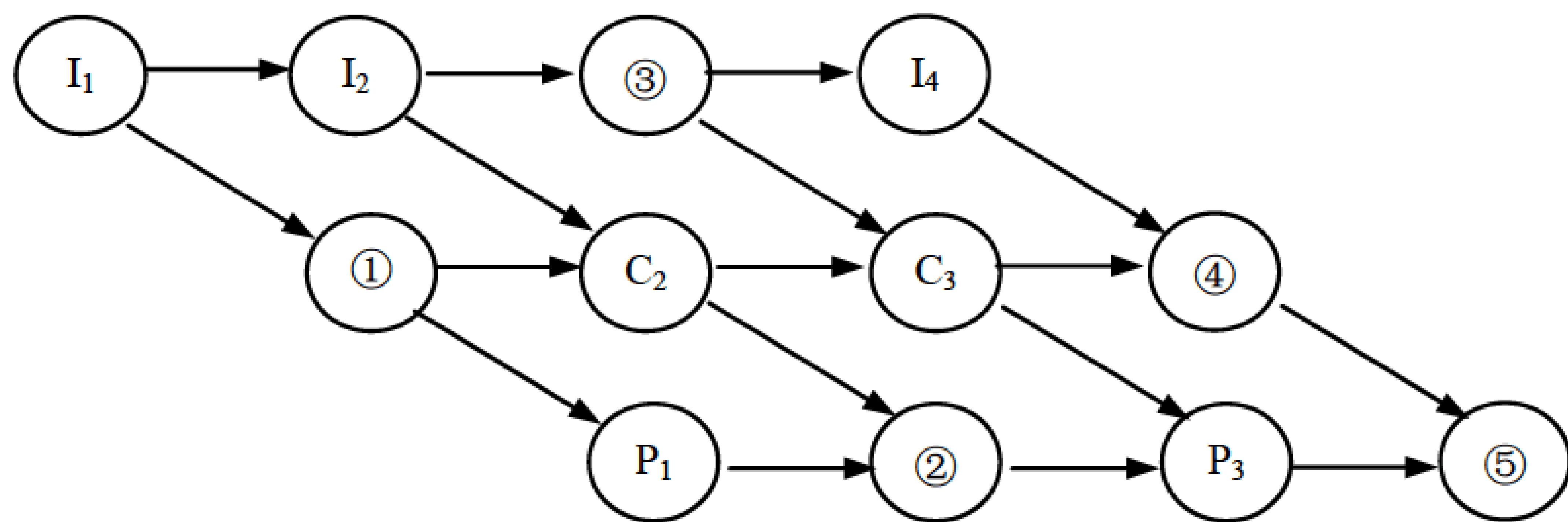
实时是指计算机对于外来信息能够以足够快的速度进行处理，并在被控对象允许的时间范围内做出快速响应。因此，实时操作系统与分时操作系统的第一点区别是交互性强弱不同，分时系统交互型强，实时系统交互性弱但可靠性要求高；第二点区别是对响应时间的敏感性强，对随机发生的外部事件必须在被控制对象规定的时间做出及时响应并对其进行处理；第三点区别是系统的设计目标不同，分时系统是设计成一个多用户的通用系统，交互能力强；而实时系统大都是专用系统。

参考答案

（23）D

试题（24）、（25）

假设某计算机系统中只有一个 CPU、一台输入设备和一台输出设备，若系统中有四个作业 T1、T2、T3 和 T4，系统采用优先级调度，且 T1 的优先级＞T2 的优先级＞T3 的优先级＞T4 的优先级。每个作业 Ti 具有三个程序段：输入 I_i、计算 C_i 和输出 P_i（i = 1,2,3,4），其执行顺序为 $I_i \to C_i \to P_i$。这四个作业各程序段并发执行的前驱图如下所示。图中①、②分别为__（24）__，③、④、⑤分别为__（25）__。



（24）A. I_2、P_2　　B. I_2、C_2　　C. C_1、P_2　　D. C_1、P_3

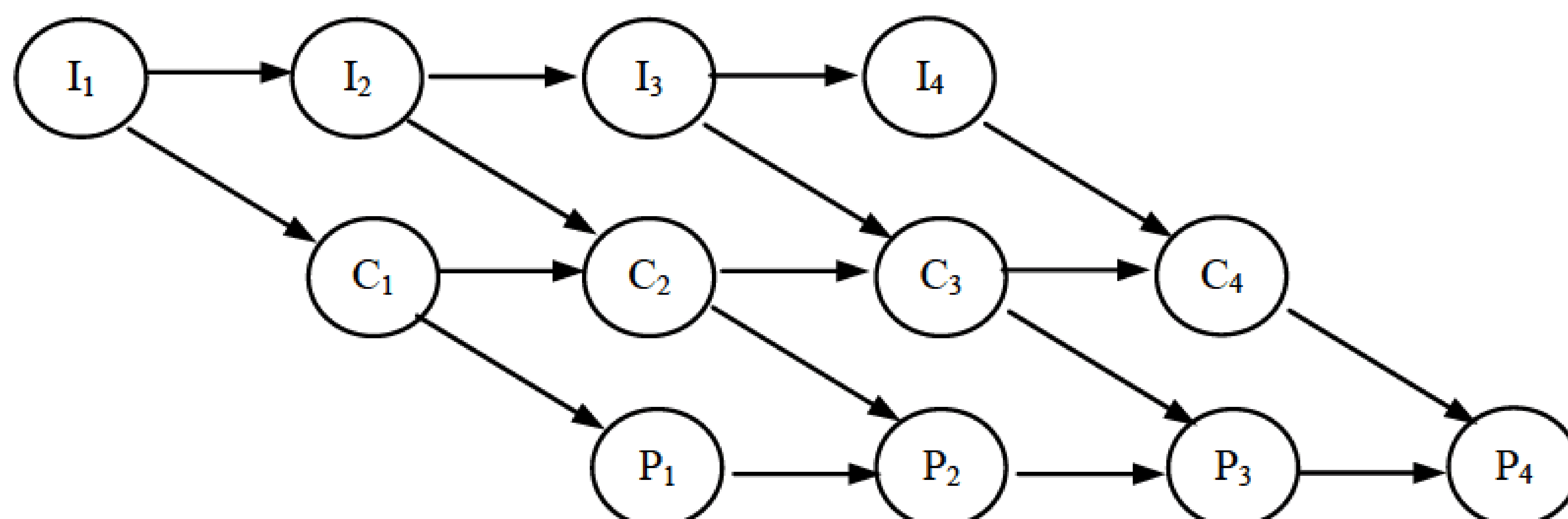
（25）A. C_2、C_4、P_4　　B. I_2、I_3、C_4　　C. I_3、P_3、P_4　　D. I_3、C_4、P_4

试题（24）、（25）分析

本题考查操作系统基础知识。

前趋图是一个有向无环图，由节点和有向边组成，节点代表各程序段的操作，而节

点间的有向边表示两个程序段操作之间存在的前趋关系（“→”）。程序段 P_i 和 P_j 的前趋关系可表示成 $P_i \rightarrow P_j$，其中 P_i 是 P_j 的前趋，P_j 是 P_i 的后继，其含义是 P_i 执行结束后 P_j 才能执行。本题完整的前趋图如下图所示，具体分析如下。



根据题意，I_1 执行结束后 C_1 才能执行，P_1 执行结束后 P_2 才能执行，因此 I_1 是 C_1 的前趋，P_1 是 P_2 的前趋。可见，①、②分别为 C_1、P_2。

根据题意，I_2 执行结束后 I_3 才能执行，即 I_2 是 I_3 的前趋，所以③应为 I_3。又因为计算机系统中只有一个 CPU 和一台输出设备，所以 C_3 执行结束后 C_4 才能执行，C_3 是 C_4 的前趋；P_3 执行结束后 P_4 才能执行，P_3 是 P_4 的前趋。经分析可知图中③、④、⑤。

参考答案

（24）C （25）D

试题（26）

假设段页式存储管理系统中的地址结构如下图所示，则系统（26）。

31　　　　　　24	23　　　　　　13	12　　　　　　0
段　号	页　号	页内地址

（26）A．最多可有 256 个段，每个段的大小均为 2048 个页，页的大小为 8K

B．最多可有 256 个段，每个段最大允许有 2048 个页，页的大小为 8K

C．最多可有 512 个段，每个段的大小均为 1024 个页，页的大小为 4K

D．最多可有 512 个段，每个段最大允许有 1024 个页，页的大小为 4K

试题（26）分析

本题考查操作系统页式存储管理方面的基础知识。从图中可见，页内地址的长度是 13 位，$2^{13}=8192$，即 8K；页号部分的地址长度是 11 位，每个段最大允许有 $2^{11}=2048$ 个页；段号部分的地址长度是 8 位，$2^8=256$，最多可有 256 个段。

参考答案

（26）B

试题（27）

假设系统中有 n 个进程共享 3 台扫描仪，并采用 PV 操作实现进程同步与互斥。若系统信号量 S 的当前值为–1，进程 P_1、P_2 又分别执行了 1 次 P（S）操作，那么信号量 S

的值应为 (27) 。

（27）A．3　　B．–3　　C．1　　D．–1

试题（27）分析

本题考查操作系统 PV 操作方面的基础知识。

系统采用 PV 操作实现进程同步与互斥，若有 n 个进程共享 3 台扫描仪，那么信号量 S 初值应为 3。若系统当前信号量 S 的值为–1，此时，P_1、P_2 又分别执行了 1 次 P（S）操作，那么当 P_1 进程执行 P（S）操作时，信号量 S 的值减 1 后等于–2；当 P_2 进程执行 P（S）操作时，信号量 S 的值减 1 后等于–3。

参考答案

（27）B

试题（28）

某字长为 32 位的计算机的文件管理系统采用位示图（bitmap）记录磁盘的使用情况。若磁盘的容量为 300GB，物理块的大小为 1MB，那么位示图的大小为 (28) 个字。

（28）A．1200　　B．3200　　C．6400　　D．9600

试题（28）分析

本题考查操作系统文件管理方面的基础知识。

根据题意，若磁盘的容量为 300GB，物理块的大小为 1MB，则该磁盘的物理块数为 300*1024=307200 个，位示图的大小为 307200/32=9600 个字。

参考答案

（28）D

试题（29）、（30）

某开发小组欲为一公司开发一个产品控制软件，监控产品的生产和销售过程，从购买各种材料开始，到产品的加工和销售进行全程跟踪。购买材料的流程、产品的加工过程以及销售过程可能会发生变化。该软件的开发最不适宜采用 (29) 模型，主要是因为这种模型 (30) 。

（29）A．瀑布　　B．原型　　C．增量　　D．喷泉

（30）A．不能解决风险　　B．不能快速提交软件

C．难以适应变化的需求　　D．不能理解用户的需求

试题（29）、（30）分析

本题考查软件开发过程模型的基础知识。

瀑布模型将开发阶段描述为从一个阶段瀑布般地转换到另一个阶段的过程。

原型模型中，开发人员快速地构造整个系统或者系统的一部分以理解或澄清问题。

增量模型是把软件产品作为一系列的增量构件来设计、编码、集成和测试，每个构件由多个相互作用的模块组成，并且能够完成特定的功能。

喷泉模型开发过程章中以用户需求为动力，以对象为驱动，适合于面向对象的开发

方法。

在上述几种开发过程模型中，瀑布模型不能适应变化的需求。

参考答案

（29）A　（30）C

试题（31）

__（31）__不属于软件质量特性中的可移植性。

（31）A．适应性　B．易安装性　C．易替换性　D．易理解性

试题（31）分析

本题考查软件质量的基础知识。

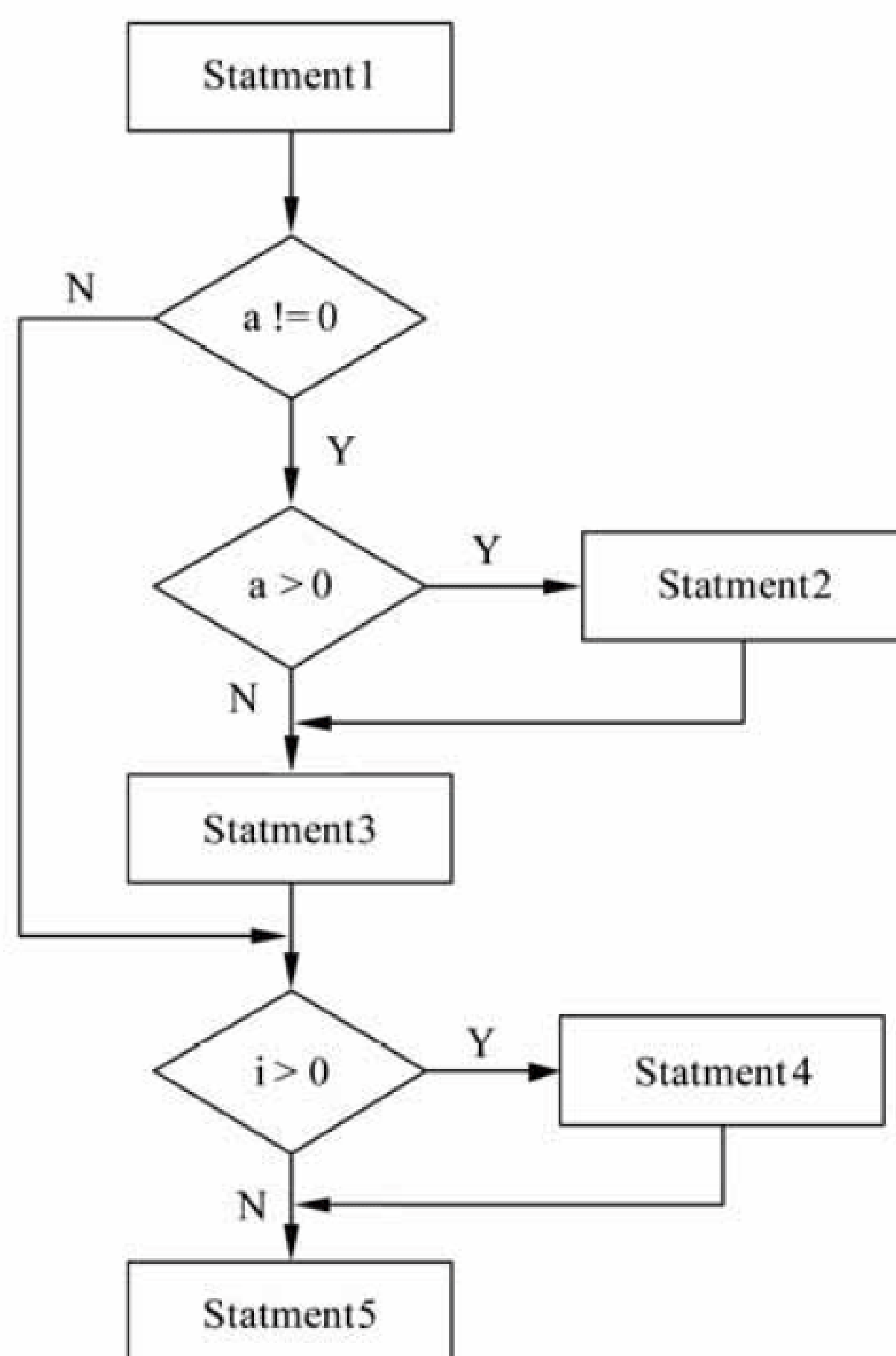
ISO/IEC 软件质量模型定义了六个软件质量特性，即功能性、可靠性、易使用性、效率、可维护性和可移植性。对每个质量特性定义其子特性。其中可移植性包括子特性：适应性、易安装性、一致性和易替换性。

参考答案

（31）D

试题（32）、（33）

对下图所示流程图采用白盒测试方法进行测试，若要满足路径覆盖，则至少需要__（32）__个测试用例。采用 McCabe 度量法计算该程序的环路复杂性为__（33）__。



（32）A．3　　B．4　　C．6　　D．8

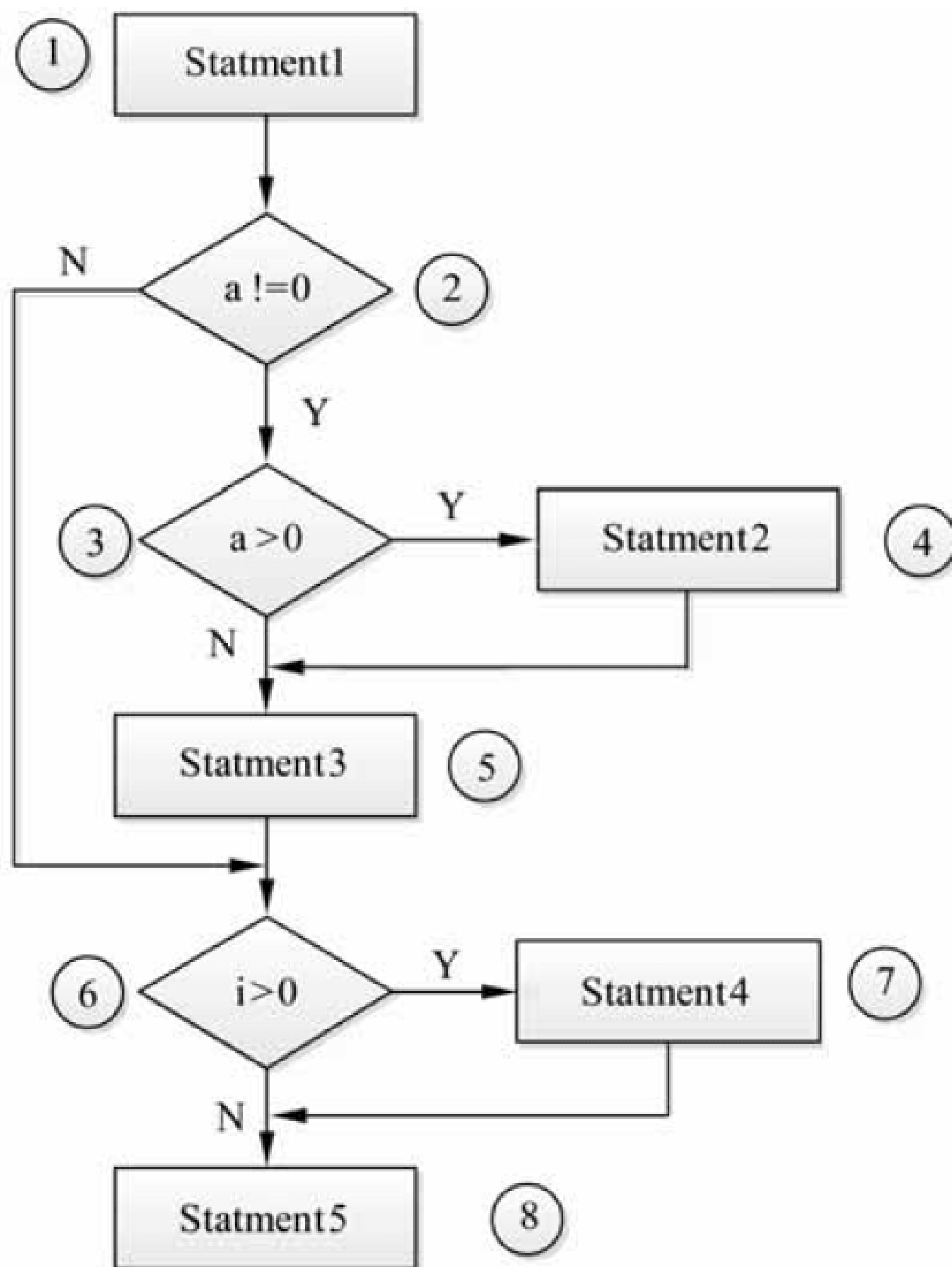
（33）A．1　　B．2　　C．3　　D．4

试题（32）、（33）分析

本题考查软件测试的基础知识。

白盒测试和黑盒测试是两种最常用的软件测试方法。路径覆盖是白盒测试的一种具体方法。

路径覆盖是指设计若干个测试用例，覆盖程序中的所有路径。



路径覆盖应使程序中每一条可能的路径至少执行一次。该流程图中一共有六条路径：①②③④⑤⑥⑦⑧，①②③④⑤⑥⑧，①②③⑤⑥⑦⑧，①②③⑤⑥⑧，①②⑥⑦⑧，①②⑥⑧，因此，实现路径覆盖至少需要 6 个测试用例。

McCabe 度量法是一种基于程序控制流的复杂性度量方法，环路复杂性为 $V(G) = m - n + 2$，图中 m=10，n=8，$V(G) = 10-8+2=4$。

参考答案

（32）C　（33）D

试题（34）

计算机系统的__（34）__可以用 MTBF/(1+MTBF)来度量，其中 MTBF 为平均失效间隔时间。

（34）A．可靠性　　B．可用性　　C．可维护性　　D．健壮性

试题（34）分析

本题考查软件质量基础知识。

可靠性、可用性和可维护性是软件的质量属性，软件工程中，用 0–1 之间的数来度量。

可靠性是指一个系统对于给定的时间间隔内、在给定条件下无失效运作的概率。可以用 MTTF/（1+MTTF）来度量，其中 MTTF 为平均无故障时间。

可用性是在给定的时间点上，一个系统能够按照规格说明正确运作的概率。可以用 MTBF/（1+MTBF）来度量，其中 MTBF 为平均失效间隔时间。

可维护性是在给定的使用条件下，在规定的时间间隔内，使用规定的过程和资源完成维护活动的概率。可以用 1/（1+MTTR）来度量，其中 MTTR 为平均修复时间。

参考答案

（34）B

试题（35）

以下关于软件测试的叙述中，不正确的是__（35）__。

（35）A．在设计测试用例时应考虑输入数据和预期输出结果
B．软件测试的目的是证明软件的正确性
C．在设计测试用例时，应该包括合理的输入条件
D．在设计测试用例时，应该包括不合理的输入条件

试题（35）分析

本题考查软件测试的基础知识。

选项 A、C、D 都与测试用例的基本概念相关，每个测试用例应该包含输入数据和预期输出结果。在设计测试用例时，要包含合理的输入和不合理的输入。因此，这三个选项均正确。

软件测试的目的是发现更多的错误，而不是证明软件的正确性。

参考答案

（35）B

试题（36）

某模块中有两个处理 A 和 B，分别对数据结构 X 写数据和读数据，则该模块的内聚类型为__（36）__内聚。

（36）A．逻辑　B．过程　C．通信　D．内容

试题（36）分析

本题考查软件设计的基础知识。

模块间的耦合和模块的内聚是度量模块独立性的两个准则。内聚是模块功能强度的度量，即模块内部各个元素彼此结合的紧密程度。一个模块内部各个元素之间的紧密程度越高，则其内聚性越高，模块独立性越好。模块内聚类型主要有以下几类：

偶然内聚，巧合内聚：指一个模块内的各处理元素之间没有任何联系。

逻辑内聚：指模块内执行若干个逻辑上相似的功能，通过参数确定该模块完成哪一个功能。

时间内聚：把需要同时执行的动作组合在一起形成的模块。

过程内聚：指一个模块完成多个任务，这些任务必须按指定的过程执行。

通信内聚：指模块内的所有处理元素都在同一个数据结构上操作，或者各处理使用相同的输入数据或产生相同的输出数据。

顺序内聚：指一个模块中的各个处理元素都密切相关于同一个功能且必须顺序执行，前一个功能元素的输出就是下一功能元素的输入。

功能内聚：指模块内的所有元素共同作用完成一个功能，缺一不可。

本题中，两个处理 A 和 B 对相同的数据结构操作，属于通信内聚。

参考答案

（36）C

试题（37）

在面向对象方法中，不同对象收到同一消息可以产生完全不同的结果，这一现象称为__（37）__。在使用时，用户可以发送一个通用的消息，而实现的细节则由接收对象自行决定。

（37）A. 接口　　B. 继承　　C. 覆盖　　D. 多态

试题（37）分析

本题考查面向对象的基础知识。

在面向对象系统中，对象是基本的运行时实体，它既包括数据（属性），也包括作用于数据的操作（行为），访问对象的这些操作也称为接口。一组大体上相似的对象定义为一个类。一个类所包含的方法和数据描述一组对象的共同行为和属性，这些对象共享这些行为和属性。有些类之间存在一般和特殊关系，在定义和实现一个类的时候，可以在一个已经存在的类的基础上来进行，把这个已经存在的类所定义的内容作为自己的内容，并加入新的内容，这种机制就是父类和子类之间共享数据和方法的机制，即继承。在子类定义时，可以继承它的父类（或祖先类）中的属性和方法，也可以重新定义父类中已经定义的方法，其方法可以对父类中方法进行覆盖，即在原有父类接口的基础上，用适合于自己要求的实现去置换父类中的相应实现。在继承的支持下，不同对象在收到同一消息是可以产生不同的结果，这是由于对通用消息的实现细节由接收对象自行决定的缘故，这就是多态。

参考答案

（37）D

试题（38）

在面向对象方法中，支持多态的是__（38）__。

（38）A．静态分配　　B．动态分配　　C．静态类型　　D．动态绑定

试题（38）分析

本题考查面向对象的基本知识。

多态的实现受到继承的支持，利用类的继承的层次关系，把具有通用功能的消息存放在高层次，而不同的实现这一功能的行为放在较低层次。当一个对象发送通用消息请求服务时，要根据接收对象的具体情况将请求的操作与实现的方法进行连接，即动态绑定，以实现在这些低层次上生成的对象给通用消息以不同的响应。

参考答案

（38）D

试题（39）

面向对象分析的目的是为了获得对应用问题的理解，其主要活动不包括__（39）__。

（39）A．认定并组织对象　　B．描述对象间的相互作用

C．面向对象程序设计　　D．确定基于对象的操作

试题（39）分析

本题考查面向对象的基本知识。

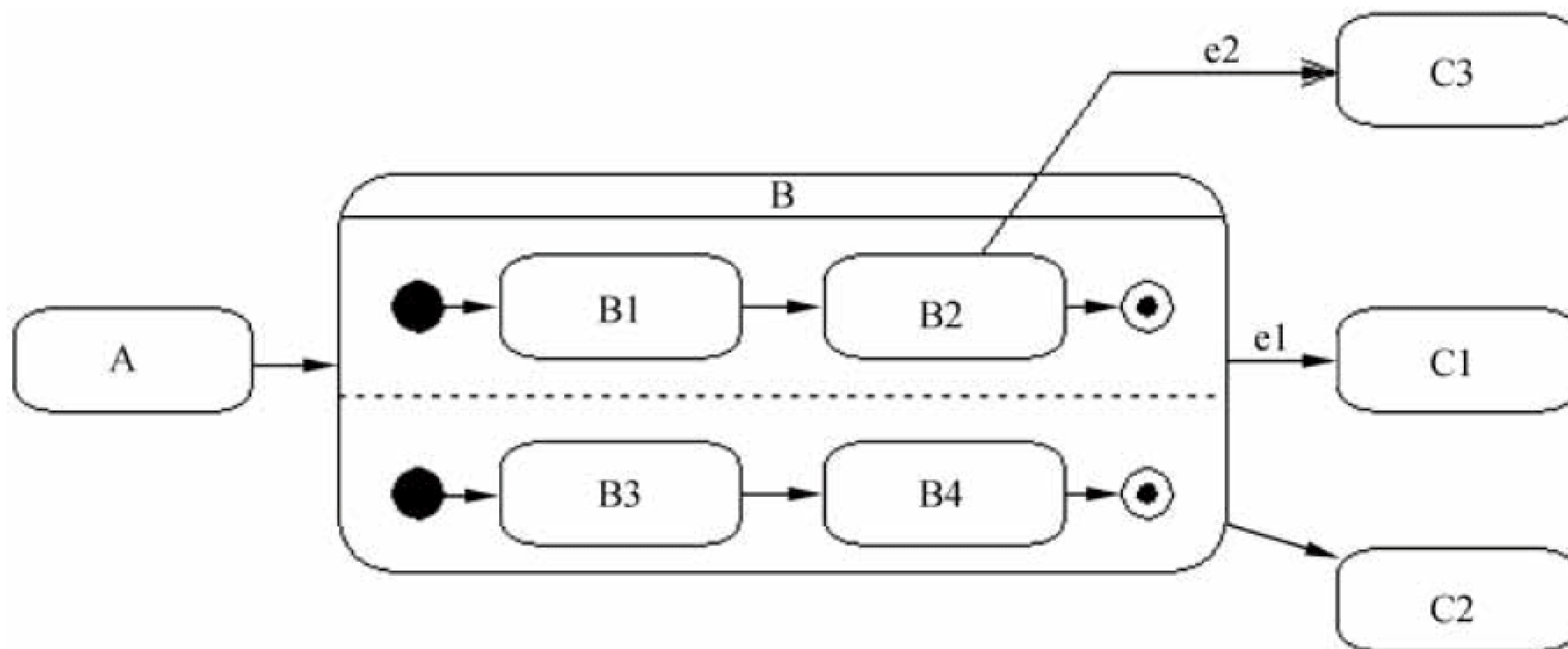
面向对象分析的目的是为了获得对应用问题的理解，以确定系统的功能、性能要求。面向对象分析方法是将数据和功能结合在一起作为一个综合对象来考虑。面向对象分析技术可以将系统的行为和信息间的关系表示为迭代构造特征。面向对象分析包含 5 个活动：认定对象、组织对象、描述对象间的相互作用、定义对象的操作、定义对象的内部信息。

参考答案

（39）C

试题（40）

如下所示的 UML 状态图中，__（40）__时，不一定会离开状态 B。



（40）A．状态 B 中的两个结束状态均达到

　　B．在当前状态为 B2 时，事件 e2 发生
　　C．事件 e2 发生
　　D．事件 e1 发生

试题（40）分析

本题考查面向对象和统一建模语言（UML）的基础知识。

状态图（state diagram）展现了一个状态机，用于描述一个对象在其生存期间的动态行为，表现为一个对象所经历的状态序列，它由状态、转换、事件和活动组成。状态图关注系统的动态视图，它对于接口、类和协作的行为建模尤为重要，强调对象行为的事件顺序。状态图通常包括简单状态和组合状态、转换（事件和动作）。

可以用状态图对系统的动态方面建模。这些动态方面可以包括出现在系统体系结构的任何视图中的任何一种对象的按事件排序的行为，这些对象包括类（各主动类）、接口、构件和节点。

当对象处于某个状态时，这个状态被称为激活状态（active state）。任何从激活状态出发的转换所标识的事件被检测到发生时，进行转换，而从当前状态出发的事件如果没有标注所检测到的事件名称，就忽略该事件，不激发任何转换，当前状态仍然是激活状态。

本题叙述中图示状态 B 内嵌套了 B1、B2、B3 和 B4。当激活状态是 B 且内嵌为状态 B2 时，如果发生事件 e2，则转移到 C3 状态；如果当前激活状态 B 的子状态不是 B2，则事件 e2 发生后，不激发状态转换。当激活状态为 B 时，不论内嵌状态是哪个，则发生事件 e1 后，激活状态转换到 C1；或者 B 中内嵌的两个结束状态均达到时，会离开状态 B。

参考答案

（40）C

试题（41）

以下关于 UML 状态图中转换（transition）的叙述中，不正确的是＿（41）＿。

（41）A．活动可以在转换时执行也可以在状态内执行
　　B．监护条件只有在相应的事件发生时才进行检查
　　C．一个转换可以有事件触发器、监护条件和一个状态
　　D．事件触发转换

试题（41）分析

本题考查面向对象和统一建模语言（UML）的基础知识。

状态图（state diagram）展现了一个状态机，关注系统的动态视图，强调对象行为的事件顺序引起的对象状态变化。

一般情况下，活动可以在状态转换时执行，也可以走状态内执行。检测到一个事件可能导致对象从一个状态移动到另一个状态，这样的移动即为转换，即事件触发转换，

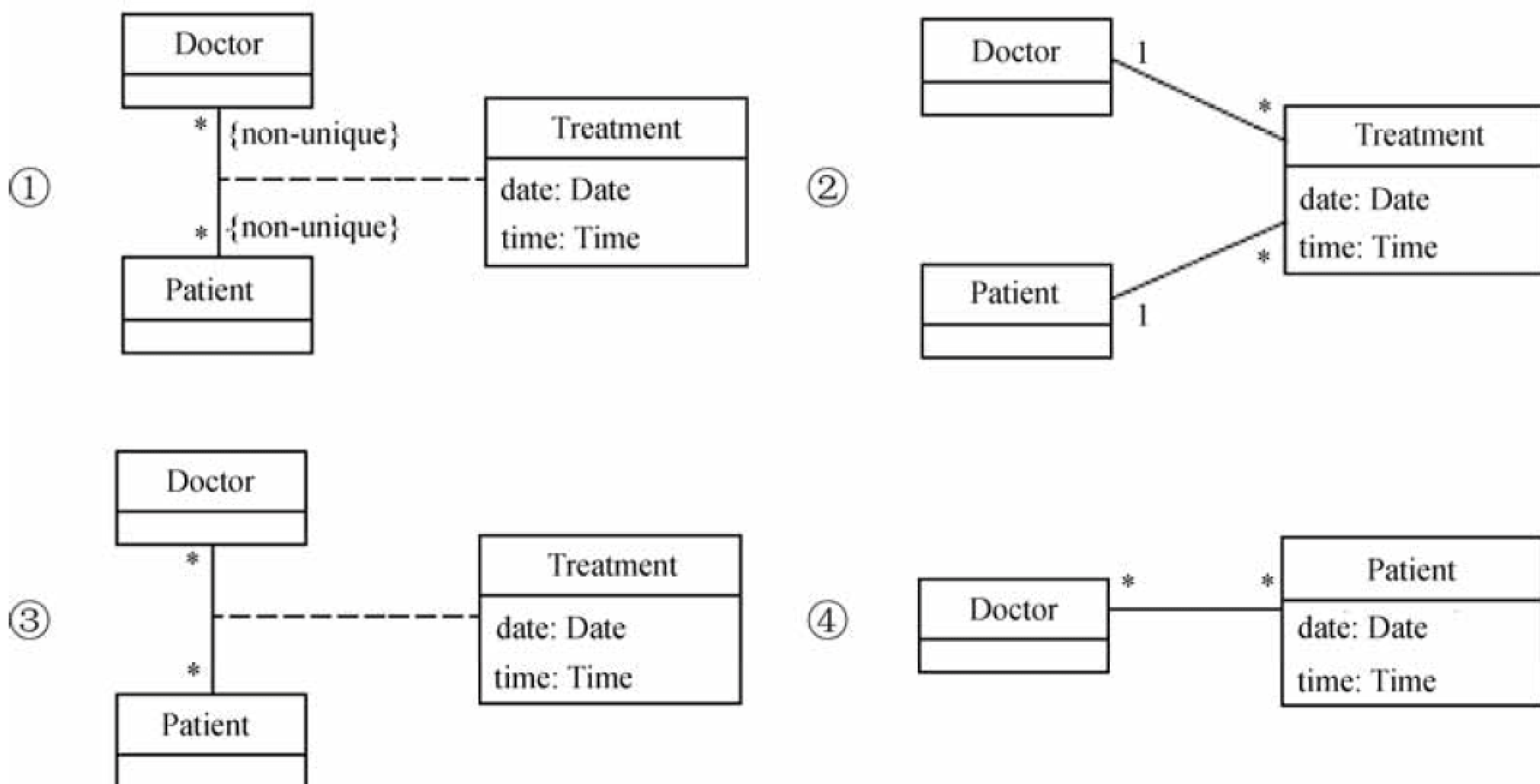
这样能引起转换的事件称为触发器。事件发生时，检查监护条件，如果满足相应的事件，则进行相应的转换，如果都没满足，则此事件没有引起状态的改变。

参考答案

（41）C

试题（42）、（43）

下图①②③④所示是 UML__（42）__。现有场景：一名医生（Doctor）可以治疗多位病人（Patient），一位病人可以由多名医生治疗，一名医生可能多次治疗同一位病人。要记录哪名医生治疗哪位病人时，需要存储治疗（Treatment）的日期和时间。以下①②③④图中__（43）__是描述此场景的模型。



（42）A．用例图　B．对象图　C．类图　D．协作图

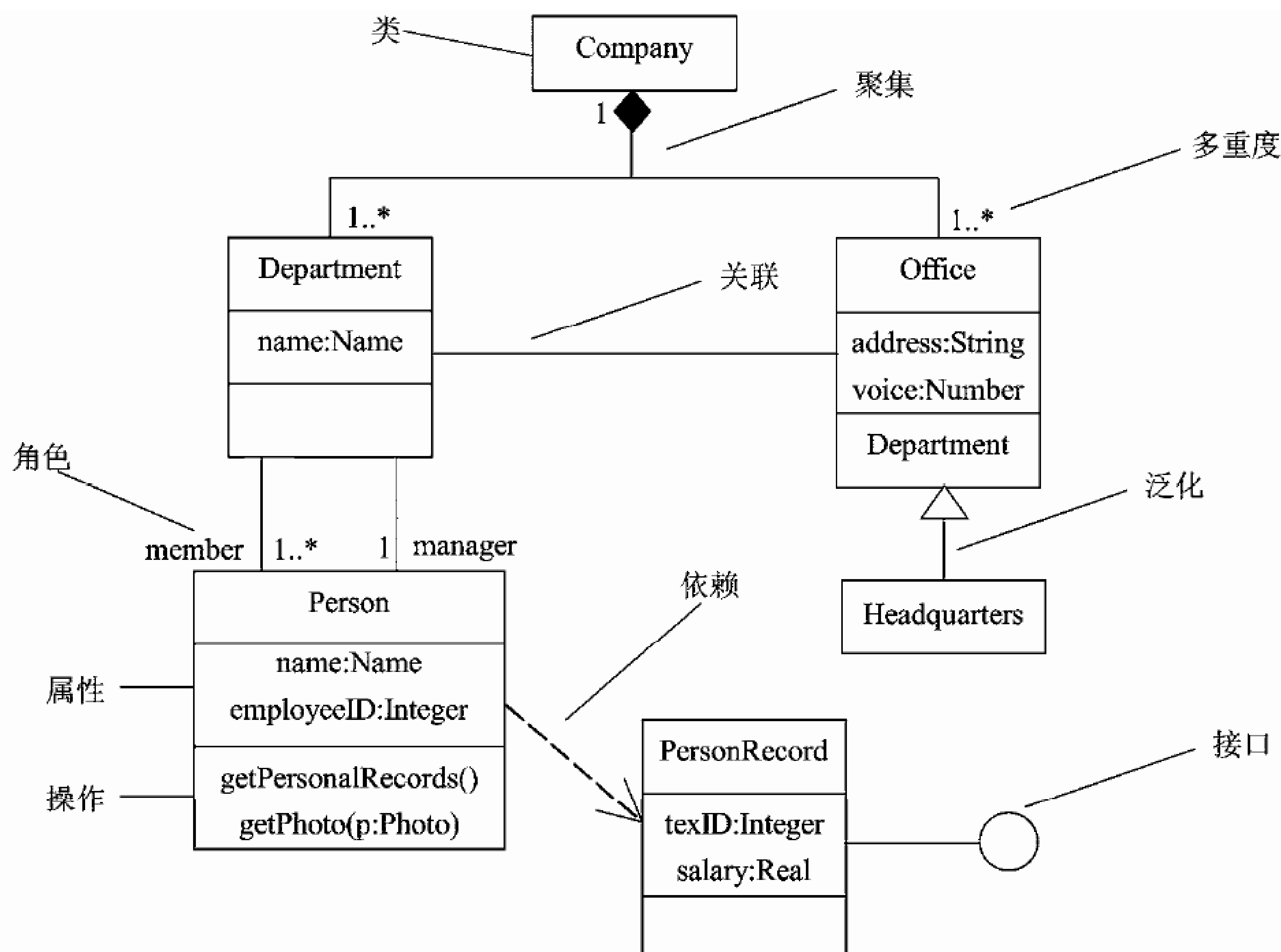
（43）A．①　B．②　C．③　D．④

试题（42）、（43）分析

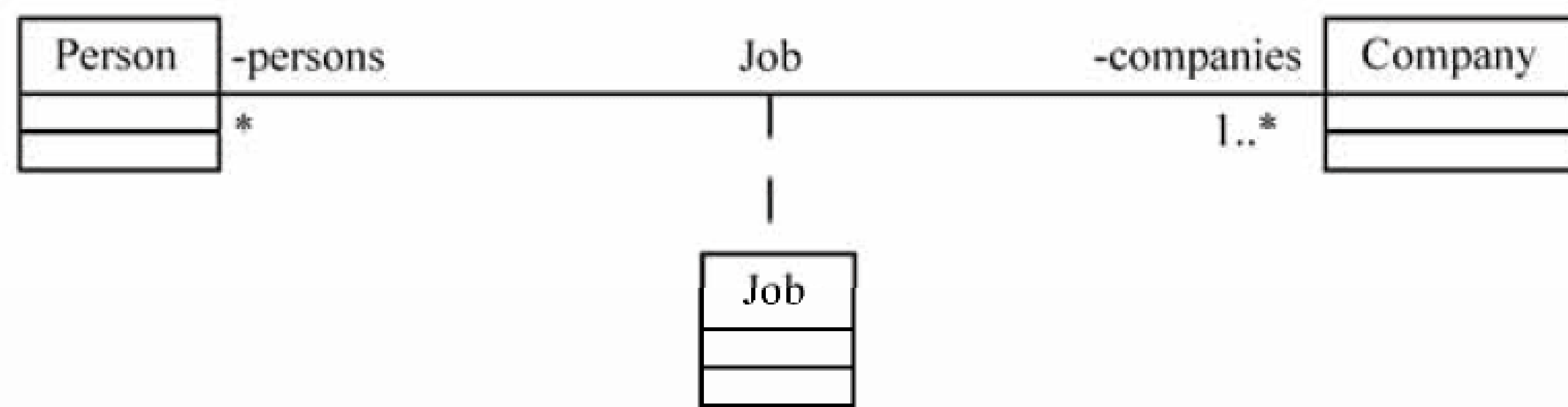
本题考查统一建模语言（UML）的基础知识。

一个类定义了一组大体上相似的对象。一个类所包含的方法和数据描述一组对象的共同行为和属性。把一组对象的共同特征加以抽象并存储在一个类中的能力，是面向对象技术最重要的一点。类图（class diagram）展现了一组对象、接口、协作和它们之间的关系。在面向对象系统的建模中所建立的最常见的图就是类图。类图给出系统的静态设计视图。包含主动类的类图给出了系统的静态进程视图。

类图中通常包括类、接口、协作、依赖、泛化和关联关系等内容（如下图所示）。类图中也可以包含注解和约束。类图还可以含有包或子系统，二者都用于把模型元素聚集成更大的组块。



在关联关系中，还可能拥有一些特性，构成类特性，即可看作是一个拥有关联特性的类，该关系兼具关联和类的特色。它定义了用于连接一些分类器，还定义属于关联关系本身的特性，这些特性只属于关联关系本身。例如要建模员工（Person）和公司（Company）之间的工作关系，有一个重要的属性是工作岗位及其岗位工资。如果将岗位工资属性放在 Person 类和 Company 类都不合适，这一属性应该放在关联关系上，这样就需要建模一个关联类 Job，用来设置岗位和岗位工资。



本题叙述中，一名医生（Doctor）可以治疗多位病人（Patient），一位病人可以由多名医生治疗，这样，医生类和病人类之间的关联关系的两端多重度均为多（*）。另外，一名医生可能多次治疗同一位病人，那么，要记录哪名医生治疗哪位病人时，需要存储治疗（Treatment）的日期和时间。这一治疗日期和时间属性放在医生类和病人类都不合适，所以这一属性应该放在关联关系上，构成关联类治疗（Treatment），并且一名医生可以多次治疗同一位病人，所以医生和病人组合并不唯一（non-unique）。

参考答案

（42）C （43）A

试题（44）、（45）

（44） 模式定义一系列的算法，把它们一个个封装起来，并且使它们可以相互替换，使得算法可以独立于使用它们的客户而变化。以下 （45） 情况适合选用该模式。

① 一个客户需要使用一组相关对象

② 一个对象的改变需要改变其他对象

③ 需要使用一个算法的不同变体

④ 许多相关的类仅仅是行为有异

（44）A. 命令（Command） B. 责任链（Chain of Responsibility）

C. 观察者（Observer） D. 策略（Strategy）

（45）A. ①② B. ②③ C. ③④ D. ①④

试题（44）、（45）分析

本题考查设计模式的基本概念。每种设计模式都有特定的意图和适用情况。

命令（Command）将一个请求封装为一个对象，从而使得可以用不同的请求对客户进行参数化；对请求排队或记录请求日志，以及支持可撤销的操作。命令模式适用于以下几种情况：

① 抽象出待执行的动作以参数化某对象，此模式是过程语言中的回调（callback）机制的一个面向对象的替代方式；

② 在不同的时刻指定、排列和执行请求；

③ 支持取消操作；

④ 支持修改日志，这样当系统崩溃时，这些修改可以被重做一遍；

⑤ 用构建在原语操作上的高层操作构造一个系统。

责任链（Chain of Responsibility）使多个对象都有机会处理请求，从而避免请求的发送者和接收者之间的耦合关系。将这些对象连成一条链，并沿着这条链传递该请求，直到有一个对象处理它为止。责任链模式适用于以下几种情况：

① 有多个的对象可以处理一个请求，哪个对象处理该请求在运行时刻自动确定；

② 在不明确指定接收者的情况下，向多个对象中的一个提交一个请求；

③ 可处理一个请求的对象集合应被动态指定。

观察者（Observer）模式定义对象间的一种一对多的依赖关系，当一个对象的状态发生改变时，所有依赖于它的对象都得到通知并被自动更新。观察者适用于以下几种情况：

① 当一个抽象模型有两个方面，其中一个方面依赖于另一个方面，将这两者封装在独立地对象中以使它们可以各自独立地改变和复用；

② 当对一个对象的改变需要同时改变其他对象，而不知道具体有多少对象有待改

变时；

③ 当一个对象必须通知其他对象，而它又不能假定其他对象是谁，即：不希望这些对象是紧耦合的。

策略（Strategy）定义一系列的算法，把它们一个个封装起来，并且使它们可以相互替换。此模式使得算法可以独立于使用它们的客户而变化。策略模式适用于以下几种情况：

① 许多相关的类仅仅是行为有异。“策略”提供了一种用多个行为中的一个行为来配置一个类的方法；

② 需要使用一个算法的不同变体。例如，定义一些反应不同空间的空间/时间权衡的算法。当这些变体实现为一个算法的类层次时，可以使用策略模式；

③ 算法使用客户不应该知道的数据。可使用策略模式以避免暴露复杂的、与算法相关的数据结构；

④ 一个类定义了多种行为，并且这些行为在这个类的操作中以多个条件语句的形式出现，将相关的条件分支移入它们各自的 Strategy 类中，以代替这些条件语句。

参考答案

（44）D　（45）C

试题（46）、（47）

<u>（46）</u>模式将一个复杂对象的构建与其表示分离，使得同样的构建过程可以创建不同的表示。以下<u>（47）</u>情况适合选用该模式。

① 抽象复杂对象的构建步骤

② 基于构建过程的具体实现构建复杂对象的不同表示

③ 一个类仅有一个实例

④ 一个类的实例只能有几个不同状态组合中的一种

（46）A. 生成器（Builder）　　B. 工厂方法（Factory Method）
　　　C. 原型（Prototype）　　D. 单例（Singleton）

（47）A. ①②　　B. ②③　　C. ③④　　D. ①④

试题（46）、（47）分析

本题考查设计模式的基本概念。每种设计模式都集中于一个特定的面向对象设计问题或设计要点，有特定的意图和适用情况。

生成器（Builder）模式将一个复杂对象的构建与它的表示分离，使得同样的构建过程可以创建不同的表示。生成器模式适用于以下几种情况：

① 当创建复杂对象的算法应该独立于该对象的组成部分以及它们的装配方式时；

② 当构造过程必须允许被构造的对象有不同的表示时。

工厂方法（Factory Method）定义一个用于创建对象的接口，让子类决定将哪一个类实例化，使一个类的实例化延迟到其子类。工厂方法适用于以下几种情况：

① 当一个类不知道它所必须创建的对象的类的时候；

② 当一个类希望由它的子类来指定它所创建的对象的时候；

③ 当类将创建对象的职责委托给多个帮助子类中的某一个，并且你希望将哪一个帮助子类是代理者这一信息局部化的时候。

原型（Prototype）模式用原型实例指定创建对象的种类，并且通过拷贝这个原型来创建新的对象。原型模式适用于以下几种情况：

① 当一个系统应该独立于它的产品创建、构成和表示时；

② 当要实例化的类是在运行时刻指定时，例如，通过动态装载；

③ 为了避免创建一个与产品类层次平行的工厂类层次时；

④ 当一个类的实例只能有几个不同状态组合中的一种时，建立相应数目的原型并克隆它们可能比每次用合适的状态手工实例化该类更方便一些。

单例（Singleton）设计模式是一种创建型模式，其意图是保证一个类仅有一个实例，并提供一个访问这个唯一实例的全局访问点。单例模式适用于以下情况：

① 当类只能有一个实例而且客户可以从一个众所周知的访问点访问它时；

② 当这个唯一实例应该是通过子类化可扩展的，并且客户应该无须更改代码就能使用一个扩展的实例时。

参考答案

（46）A （47）A

试题（48）

由字符 a、b 构成的字符串中，若每个 a 后至少跟一个 b，则该字符串集合可用正规式表示为__（48）__。

（48）A. (b|ab)* B. (ab*)* C. (a*b*)* D. (a|b)*

试题（48）分析

本题考查程序语言知识。

正规式 (b|ab)表示的正规集为{b, ab}，(b|ab)*表示的正规集为{ ε ,b, ab, bb, bab, abb, abab, bbb, bbab, babb, babab, abbb, abbab, ababb, ababab, ...}，用自然语言描述就是每个 a 后面都至少有 1 个 b。

正规式 (ab*)表示的正规集为{ ε ,a, ab, abb, abbb, abbbb, ...}，(ab*)*表示的正规集为{aa, aab, aabb, aabbb, aabbbb, aba, abba, abbba, abab, abbab, ...}，用自然语言描述就是除了空串，每个串中都至少有 1 个 a。

正规式(a*b*)*和(a|b)*是等价的，它们都表示{ ε ,a, b, aa, ab, ba, bb, aaa, aab, aba, abb, baa, bab, bab, bbb, ... }，用自然语言描述就是用 a、b 构成的任何字符串。

参考答案

（48）A

试题（49）

乔姆斯基（Chomsky）将文法分为 4 种类型，程序设计语言的大多数语法现象可用其中的__（49）__描述。

（49）A．上下文有关文法　　B．上下文无关文法
　　　C．正规文法　　D．短语结构文法

试题（49）分析

本题考查程序语言知识。

程序语言的大多数语法现象可用乔姆斯基的上下文无关文法描述。

参考答案

（49）B

试题（50）

运行下面的 C 程序代码段，会出现__（50）__错误。

```
int k = 0;
for (; k < 100; );
{ k++;}
```

（50）A．变量未定义　　B．静态语义　　C．语法　　D．动态语义

试题（50）分析

本题考查程序语言知识。

代码段中“for (; k < 100;);”的循环体为空语句，循环条件中的 k 值在循环中没有改变，因此“k < 100”是一直成立的，此代码段是无限循环的，只有运行时才能表现出来，属于动态语义错误。

参考答案

（50）D

试题（51）

在数据库系统中，一般由 DBA 使用 DBMS 提供的授权功能为不同用户授权，其主要目的是为了保证数据库的__（51）__。

（51）A．正确性　　B．安全性　　C．一致性　　D．完整性

试题（51）分析

本题考查数据库安全控制方面的基础知识。

数据库管理系统的安全措施有 3 个方面：

① 权限机制：通过权限机制，限定用户对数据的操作权限，把数据的操作限定在具有指定权限的用户范围内，以保证数据的安全。在标准 SQL 中定义了授权语句 GRANT 来实现权限管理。

② 视图机制：通过建立用户视图，用户或应用程序只能通过视图来操作数据，保

证了视图之外的数据的安全性。

③ 数据加密：对数据库中的数据进行加密，可以防止数据在存储和传输过程中失密。

参考答案

（51）B

试题（52）、（53）

给定关系模式 $R(U, F)$，其中：U 为关系模式 R 中的属性集，F 是 U 上的一组函数依赖。假设 $U=\{A_1,A_2,A_3,A_4\}$，$F=\{A_1 \to A_2, A_1A_2 \to A_3, A_1 \to A_4, A_2 \to A_4\}$，那么关系 R 的主键应为__（52）__。函数依赖集 *F* 中的__（53）__是冗余的。

（52）A. A_1　　B. A_1A_2　　C. A_1A_3　　D. $A_1A_2A_3$

（53）A. $A_1 \to A_2$　　B. $A_1A_2 \to A_3$　　C. $A_1 \to A_4$　　D. $A_2 \to A_4$

试题（52）、（53）分析

本题考查关系数据库规范化理论方面的基础知识。

根据题意，$F=\{A_1 \to A_2, A_1A_2 \to A_3, A_1 \to A_4, A_2 \to A_4\}$，不难得出属性 A_1A_2 决定全属性 U，所以 A_1A_2 为候选关键字。由于 $A_1 \to A_2$，$A_2 \to A_4$ 可以推出 $A_1 \to A_4$（传递率），所以函数依赖集 $A_1 \to A_4$ 是冗余的。

参考答案

（52）B （53）C

试题（54）、（55）

给定关系 $R(A,B,C,D)$ 和关系 $S(A,C,E,F)$，对其进行自然连接运算 $R \bowtie S$ 后的属性列为__（54）__个；与 $\sigma_{R.B>S.E}(R \bowtie S)$ 等价的关系代数表达式为__（55）__。

（54）A. 4　　B. 5　　C. 6　　D. 8

（55）A. $\sigma_{2>7}(R \times S)$　　B. $\pi_{1,2,3,4,7,8}(\sigma_{1=5 \wedge 2>7 \wedge 3=6}(R \times S))$

C. $\sigma_{2>'7'}(R \times S)$　　D. $\pi_{1,2,3,4,7,8}(\sigma_{1=5 \wedge 2>'7' \wedge 3=6}(R \times S))$

试题（54）、（55）分析

本题考查关系代数运算方面的基础知识。

自然连接是一种特殊的等值连接，它要求两个关系中进行比较的分量必须是相同的属性组，并且在结果集中去掉右边重复的属性列。对关系 R 和 S 进行自然连接运算后的属性列数为 6 个，即为 $R.A, R.B, R.C, R.D, S.E, S.F$。

对于试题（55），项 A 和 C 是错误的，因为 $R \times S$ 的结果集的属性列为 $R.A, R.B, R.C, R.D, S.A, S.C, S.E, S.F$，选取运算 σ 是对关系进行横向运算，没有去掉重复属性列。选项 B "$\pi_{1,2,3,4,7,8}(\sigma_{1=5 \wedge 2>7 \wedge 3=6}(R \times S))$" 的含义为 R 与 S 的笛卡儿积中选择第 1 个属性列=第 5 个属性列（即 R.A=S.A），同时满足第 2 个属性列＞第 7 个属性列（即 R.B＞S.E），同时满足第 3 个属性列=第 6 个属性列（即 R.C=S.C）。选项 D 错误的原因

是选取运算 $\sigma_{1=5\wedge 2>'7'\wedge 3=6}(R\times S)$ 中的条件“2>'7'”与题意不符，其含义是 R.B 的值大于 7（属性列数字 7 加了单引号表示数值 7），而不是 R.B>S.E。

参考答案

（54）C　（55）B

试题（56）

下列查询 B =“大数据”且 F=“开发平台”，结果集属性列为 A、B、C、F 的关系代数表达式中，查询效率最高的是＿（56）＿。

（56）A. $\pi_{1,2,3,8}(\sigma_{2='大数据'\wedge 1=5\wedge 3=6\wedge 8='开发平台'}(R\times S))$

B. $\pi_{1,2,3,8}(\sigma_{1=5\wedge 3=6\wedge 8='开发平台'}(\sigma_{2='大数据'}(R)\times S))$

C. $\pi_{1,2,3,8}(\sigma_{2='大数据'\wedge 1=5\wedge 3=6}(R\times \sigma_{4='开发平台'}(S))$

D. $\pi_{1,2,3,8}(\sigma_{1=5\wedge 3=6}(\sigma_{2='大数据'}(R)\times \sigma_{4='开发平台'}(S)))$

试题（56）分析

本题考查关系代数运算方面的基础知识。

关系代数表达式查询优化的原则如下：

① 提早执行选取运算。对于有选择运算的表达式，应优化成尽可能先执行选择运算的等价表达式，以得到较小的中间结果，减少运算量以及从外存读块的次数。

② 合并乘积与其后的选择运算为连接运算。在表达式中，当乘积运算后面是选择运算时，应该合并为连接运算，使选择与乘积一道完成，以避免做完乘积后，需再扫描一个大的乘积关系进行选择运算。

③ 将投影运算与其后的其他运算同时进行，以避免重复扫描关系。

④ 将投影运算和其前后的二目运算结合起来，使得没有必要为去掉某些字段再扫描一遍关系。

⑤ 在执行连接前对关系适当地预处理，就能快速地找到要连接的元组。方法有两种：索引连接法、排序合并连接法。

⑥ 存储公共子表达式。对于有公共子表达式的结果应存于外存（中间结果），这样，当从外存读出它的时间比计算的时间少时，就可节约操作时间。

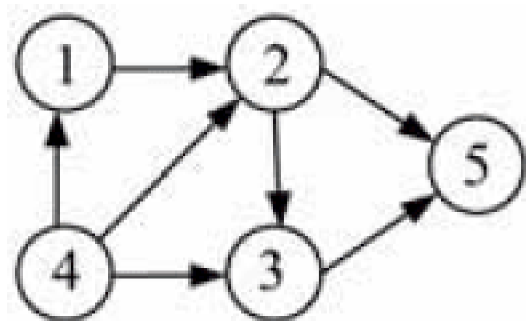
显然，根据原则①尽量提早执行选取运算。

参考答案

（56）D

试题（57）

拓扑序列是有向无环图中所有顶点的一个线性序列，若有向图中存在弧<v,w>或存在从顶点 v 到 w 的路径，则在该有向图的任一拓扑序列中，v 一定在 w 之前。下面有向图的拓扑序列是＿（57）＿。



（57）A．4 1 2 3 5　　B．4 3 1 2 5　　C．4 2 1 3 5　　D．4 1 3 2 5

试题（57）分析

本题考查数据结构基础知识。

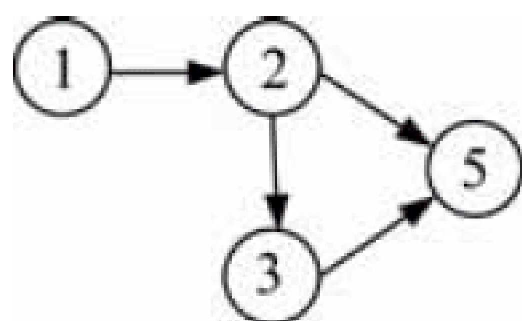
对有向无环图网进行拓扑排序的方法如下：

① 在 AOV 网中选择一个入度为零（没有前驱）的顶点 v 且输出它；

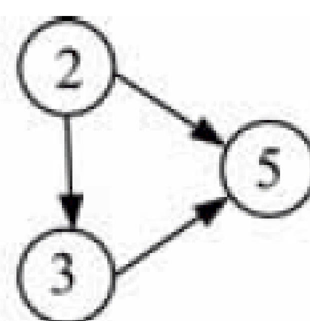
② 从网中删除该顶点 v 以及与该顶点有关的所有边；

③ 重复上述两步，直至网中不存在入度为零的顶点为止。

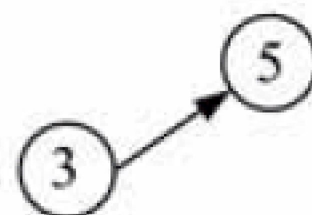
按照上述方法，拓扑序列的第一个顶点为 4，执行①和②步之后的有向图如下图（a）所示。接下来再输出的顶点只能为 1，因此执行①和②步之后的有向图如下图（b）所示。接下来再输出的顶点只能为 2，因此①和②步之后的有向图如下图（c）所示。因此，拓扑序列为 41235。



（a）



（b）



（c）

参考答案

（57）A

试题（58）、（59）

设有一个包含 n 个元素的有序线性表。在等概率情况下删除其中的一个元素，若采用顺序存储结构，则平均需要移动__（58）__个元素；若采用单链表存储，则平均需要移动__（59）__个元素。

（58）A．1　　B．(n-1)/2　　C．log n　　D．n

（59）A．0　　B．1　　C．(n-1)/2　　D．n/2

试题（58）、（59）分析

本题考查数据结构基础知识。

线性表是一个线性序列，在顺序存储方式下，若删除其中一个元素，需要将其后的元素逐个前移，使得元素之间没有空闲单元。表长为 n 时，共有 n 个可删除的元素，删除元素 a_1 时需要移动 n–1 个元素，删除元素 a_n 时不需要移动元素，因此，等概率下删除一个元素时平均的移动元素次数 E_{delete} 为

$$E_{delete} = \sum_{i=1}^{n} q_i \times (n-i) = \frac{1}{n}\sum_{i=1}^{n}(n-i) = \frac{n-1}{2}$$

线性表若采用单链表存储，插入和删除元素的实质都是对相关指针的修改，而不需

要移动元素。

参考答案

（58）B　　（59）A

试题（60）

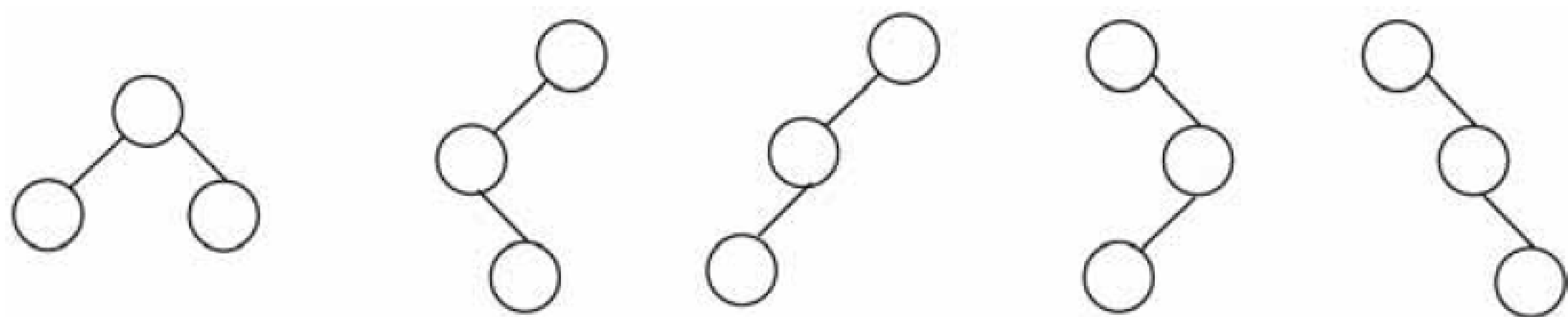
具有 3 个结点的二叉树有__（60）__种形态。

（60）A．2　　B．3　　C．5　　D．7

试题（60）分析

本题考查数据结构基础知识。

具有 3 个结点的二叉树有以下 5 种形态，如下图所示。



参考答案

（60）C

试题（61）

以下关于二叉排序树（或二叉查找树、二叉检索树）的叙述中，正确的是__（61）__。

（61）A．对二叉排序树进行先序、中序和后序遍历，都得到结点关键字的有序序列

B．含有 n 个结点的二叉排序树高度为$\lfloor \log_2 n \rfloor + 1$

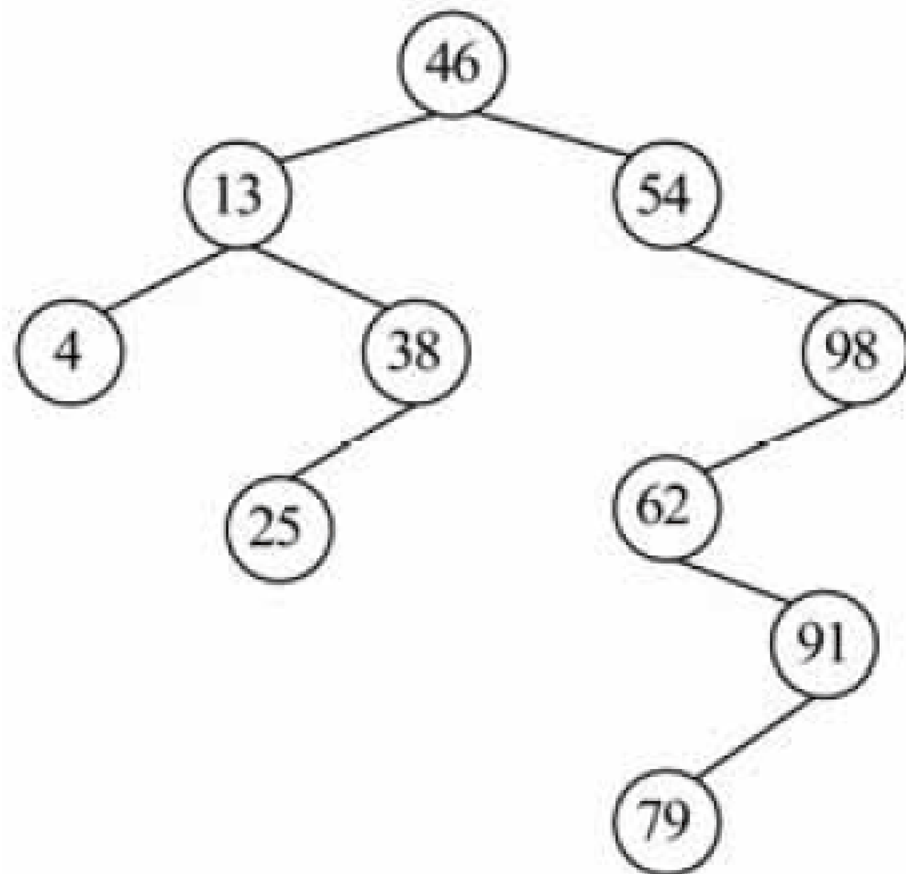
C．从根到任意一个叶子结点的路径上，结点的关键字呈现有序排列的特点

D．从左到右排列同层次的结点，其关键字呈现有序排列的特点

试题（61）分析

本题考查数据结构基础知识。

二叉查找树又称为二叉排序树或二叉检索树，它或者是一棵空树，或者是具有如下性质的二叉树：①若它的左子树非空，则左子树中所有结点的值均小于根结点的值；②若它的右子树非空，则右子树中所有结点的值均大于根结点的值；③左、右子树本身就是二叉查找树。某二叉排序树如下图所示。



以上图为例，对非空二叉排序树进行中序遍历，得到递增有序的序列，先序和后序序列则不是。因此，选项A的说法是错误的。

二叉排序树中结点在左、右子树上的分布并不均匀，极端情况下，n个结点的二叉排序树的高度为n。因此，选项B的说法是错误的。

以上图为例，从46到25的路径上的结点关键码序列为46, 13, 38, 25，并不是一个有序序列。因此，选项C的说法是错误的。

参考答案

（61）D

试题（62）、（63）

下表为某文件中字符的出现频率，采用霍夫曼编码对下列字符编码，则字符序列“bee”的编码为__（62）__；编码“110001001101”对应的字符序列为__（63）__。

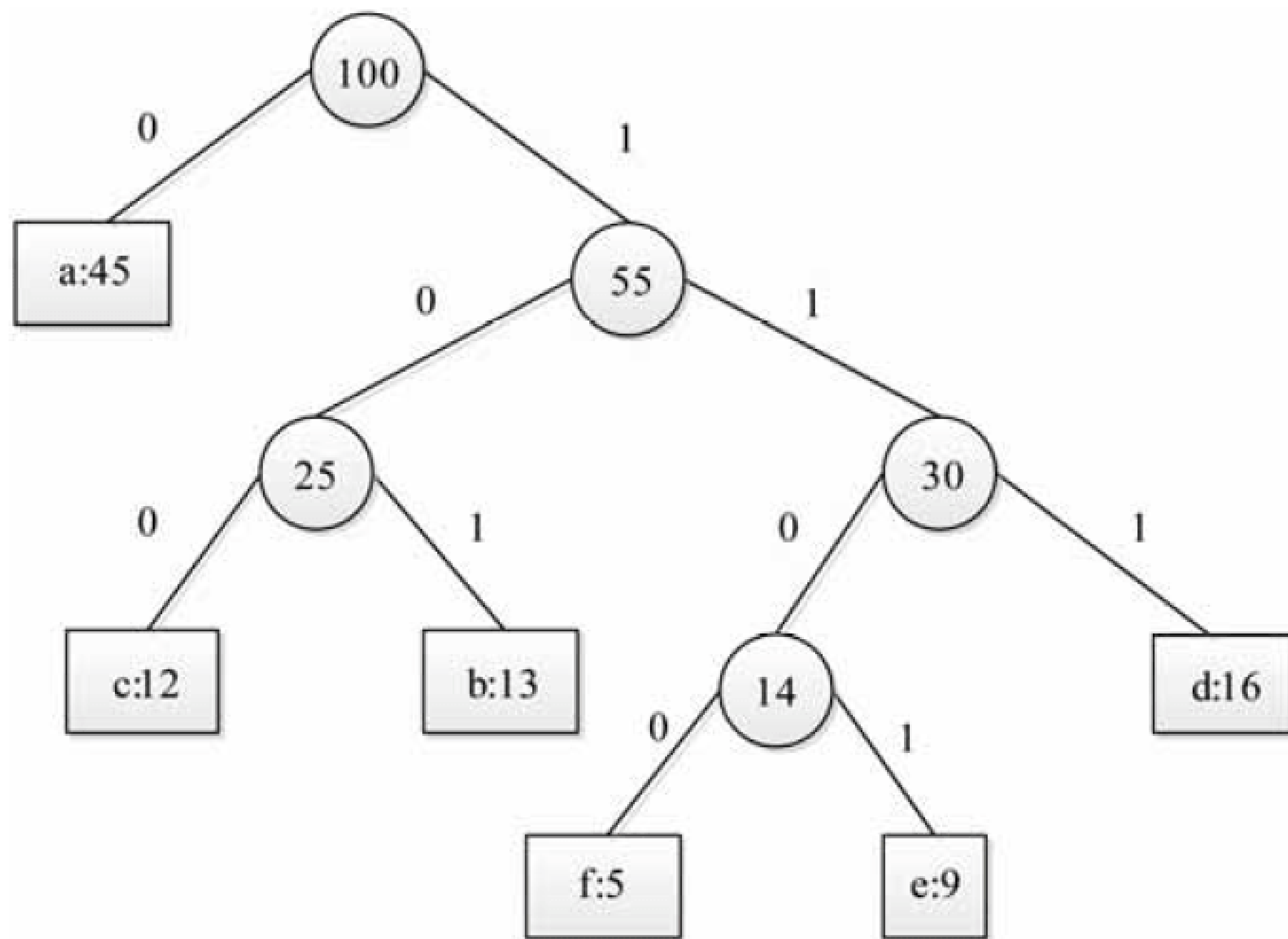
字符	a	b	c	d	e	f
频率(%)	45	13	12	16	9	5

（62）A. 10111011101　B. 10111001100　C. 001100100　D. 110011011

（63）A. bad　B. bee　C. face　D. bace

试题（62）、（63）分析

本题考查算法设计与分析的基础知识。题干中给出的实例的霍夫曼编码树如下图所示。



根据该图，bee的编码为101 1101 1101。而编码“110001001101”对应的字符序列则为face。

参考答案

（62）A （63）C

试题（64）、（65）

两个矩阵 A_{m*n} 和 B_{n*p} 相乘，用基本的方法进行，则需要的乘法次数为 m*n*p。多个矩阵相乘满足结合律，不同的乘法顺序所需要的乘法次数不同。考虑采用动态规划方法确定 M_i，$M_{(i+1)}$，…，M_j 多个矩阵连乘的最优顺序，即所需要的乘法次数最少。最少乘法次数用 m[i,j]表示，其递归式定义为：

$$m[i,j]=\begin{cases}0 & i \geqslant j \\ \min\limits_{i \leqslant k < j}\{m[i,k]+m[k+1,j]+p_{i-1}p_kp_j\} & i<j\end{cases}$$

其中，i、j 和 k 为矩阵下标，矩阵序列中 M_i 的维度为$(p_{i-1})*p_i$。采用自底向上的方法实现该算法来确定 n 个矩阵相乘的顺序，其时间复杂度为__（64）__。若四个矩阵 M_1、M_2、M_3、M_4 相乘的维度序列为 2、6、3、10、3，采用上述算法求解，则乘法次数为__（65）__。

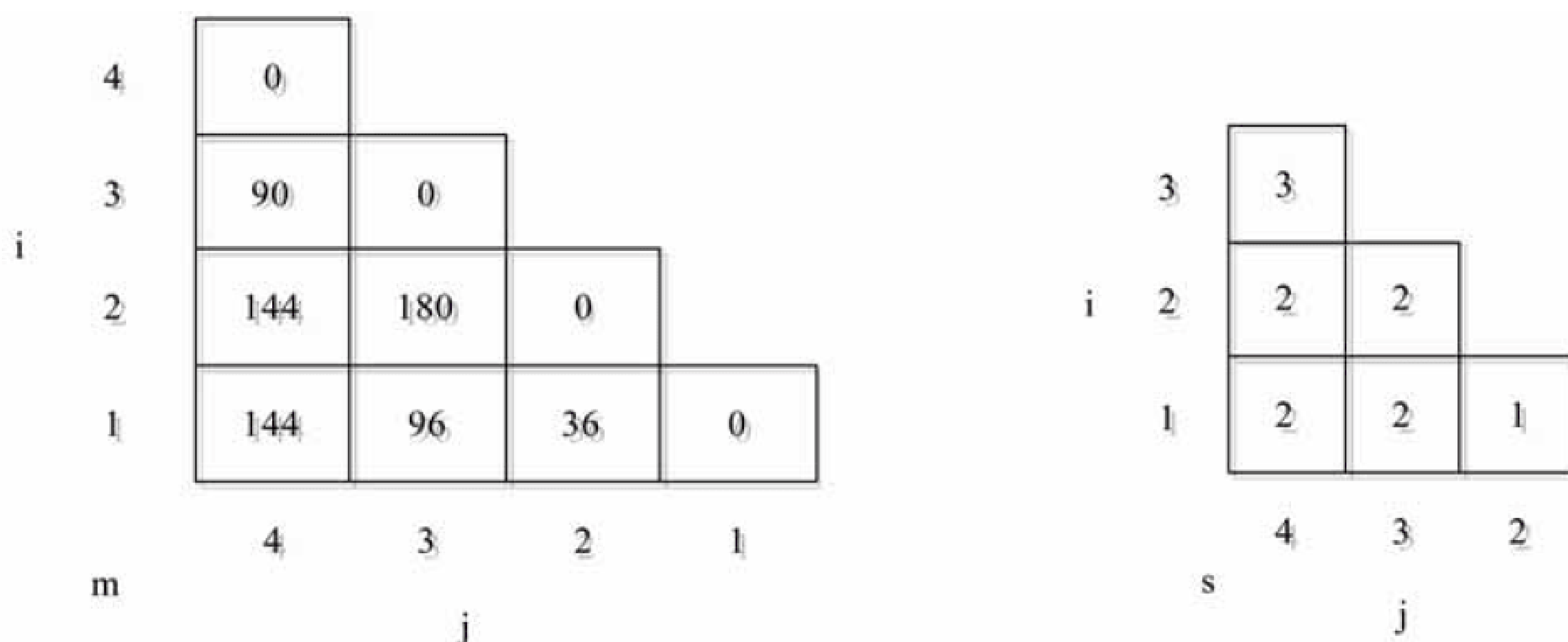
（64）A．$O(n^2)$　　B．$O(n^2lgn)$　　C．$O(n^3)$　　D．$O(n^3lgn)$

（65）A．156　　B．144　　C．180　　D．360

试题（64）、（65）分析

本题考查算法设计与分析的基础知识。

矩阵链乘是一个最优化问题，求解 n 个矩阵相乘的最优加括号方式，可以用动态规划方法来求解。题干已经给出动态规划求解的递归式。根据上式计算 m 的值，同时记录 k 的值到 s 中。



可以得到最优的加括号方式$((M_1M_2)(M_3M_4))$，乘法次数为 144。因此（65）题选择 B。

而根据该递归式自底向上求解时，应该用三重循环进行，即矩阵链长度 l 从 1 到 n，子矩阵链起始位置，即 i 从 1 到 n-l+1，矩阵链分开的位置 k，从 i 到 j-1。因此时间复杂度为 $O(n^3)$。

参考答案

（64）C　（65）B

试题（66）、（67）

以下协议中属于应用层协议的是__（66）__，该协议的报文封装在__（67）__中传送。

（66）A．SNMP　B．ARP　C．ICMP　D．X.25

（67）A．TCP　B．IP　C．UDP　D．ICMP

试题（66）、（67）分析

属于应用层协议的是简单网络管理协议 SNMP，它的传输层协议是 UDP。ARP 和 ICMP 都属于网络层协议。X.25 是分组交换网上的协议，也归于网络层。

参考答案

（66）A　（67）C

试题（68）

某公司内部使用 wb.xyz.com.cn 作为访问某服务器的地址，其中 wb 是__（68）__。

（68）A．主机名　B．协议名　C．目录名　D．文件名

试题（68）分析

本题考查 URL 的基础知识。

URL（Uniform Resource Locator，统一资源定位符）是对互联网上的资源位置和访问方法的一种简洁的表示，是互联网上资源的地址。互联网上的每个文件都有一个唯一的 URL，它包含的信息指出文件的位置以及浏览器应该怎么处理它。

一个标准 URL 的格式如下：

协议://主机名.域名.域名后缀或 IP 地址（:端口号）/目录/文件名

其中，目录可能是多级的。

参考答案

（68）A

试题（69）

如果路由器收到了多个路由协议转发的关于某个目标的多条路由，那么决定采用哪条路由的策略是__（69）__。

（69）A．选择与自己路由协议相同的　B．选择路由费用最小的

C．比较各个路由的管理距离　D．比较各个路由协议的版本

试题（69）分析

各种路由来源的管理距离如下表所示。

路由来源	管理距离	路由来源	管理距离
直连路由	0	IS-IS	115
静态路由	1	RIP	120
EIGRP 汇总路由	5	EGP	140
外部 BGP	20	ODR（按需路由）	160
内部 EIGRP	90	外部 EIGRP	170
IGRP	100	内部 BGP	200
OSPF	110	未知	255

如果路由器收到了由多个路由协议转发的、关于某个目标的多条路由，则比较各个路由的管理距离，并采用管理距离小的路由来源提供的路由信息。

参考答案

（69）C

试题（70）

与地址 220.112.179.92 匹配的路由表的表项是＿（70）＿。

（70）A．220.112.145.32/22　　B．220.112.145.64/22

C．220.112.147.64/22　　D．220.112.177.64/22

试题（70）分析

地址 220.112.145.32/22 的二进制形式是 **1101 1100. 0111 0000. 1001 00**01. 0010 0000

地址 220.112.145.64/22 的二进制形式是 **1101 1100. 0111 0000. 1001 00**01. 0100 0000

地址 220.112.147.64/22 的二进制形式是 **1101 1100. 0111 0000. 1001 00**11. 0100 0000

地址 220.112.177.64/22 的二进制形式是 **1101 1100. 0111 0000. 1011 00**01. 0100 0000

而地址 220.112.179.92 的二进制形式是 **1101 1100. 0111 0000. 1011 00**11. 0101 1100

所以与地址 220.112.179.92 匹配的是 220.112.177.64/22。

参考答案

（70）D

试题（71）～（75）

Software entities are more complex for their size than perhaps any other human construct, because no two parts are alike (at least above the statement level). If they are, we make the two similar parts into one, a ＿（71）＿, open or closeD. In this respect software systems differ profoundly from computers, buildings, or automobiles, where repeated elements abound.

Digital computers are themselves more complex than most things people build; they have very large numbers of states. This makes conceiving, describing, and testing them harD. Software systems have orders of magnitude more ＿（72）＿ than computers do.

Likewise, a scaling-up of a software entity is not merely a repetition of the same elements in larger size; it is necessarily an increase in the number of different elements. In most cases, the elements interact with each other in some ＿（73）＿ fashion, and the complexity of the whole increases much more than linearly.

The complexity of software is a(an) ＿（74）＿ property, not an accidental one. Hence descriptions of a software entity that abstract away its complexity often abstract away its essence. Mathematics and the physical sciences made great strides for three centuries by constructing simplified models of complex phenomena, deriving properties from the models, and verifying those properties experimentally. This worked because the complexities＿（75）＿ in the models were not the essential properties of the phenomenA. It does not work when the

complexities are the essence.

Many of the classical problems of developing software products derive from this essential complexity and its nonlinear increases with size. Not only technical problems but management problems as well come from the complexity.

（71）A. task B. job C. subroutine D. program

（72）A. states B. parts C. conditions D. expressions

（73）A. linear B. nonlinear C. parallel D. additive

（74）A. surface B. outside C. exterior D. essential

（75）A. fixed B. included C. ignored D. stabilized

参考译文

规模上，软件实体可能比任何由人类创造的其他实体要复杂，因为没有任何两个软件部分是相同的（至少是在语句的级别）。如果有相同的情况，我们会把它们合并成供调用的子函数。在这个方面，软件系统与计算机、建筑或者汽车大不相同，后者往往存在着大量重复的部分。

数字计算机本身就比人类建造的大多数东西复杂。计算机拥有大量的状态，这使得构思、描述和测试都非常困难。软件系统的状态又比计算机系统状态多若干个数量级。

同样，软件实体的扩展也不仅仅是相同元素重复添加，而必须是不同元素实体的添加。大多数情况下，这些元素以非线性递增的方式交互，因此整个软件的复杂度以更大的非线性级数增长。

软件的复杂度是必要属性，不是次要因素。因此，抽掉复杂度的软件实体描述常常也去掉了一些本质属性。数学和物理学在过去三个世纪取得了巨大的进步，数学家和物理学家们建立模型以简化复杂的现象，从模型中抽取出各种特性，并通过试验来验证这些特性。这些方法之所以可行——是因为模型中忽略的复杂度不是被研究现象的必要属性。当复杂度是本质特性时，这些方法就行不通了。

上述软件特有的复杂度问题造成了很多经典的软件产品开发问题。复杂度不仅仅导致技术上的困难，还引发了很多管理上的问题。

参考答案

（71）C （72）A （73）B （74）D （75）C

第 32 章　2016 下半年软件设计师下午试题分析与解答

试题一（共 15 分）

阅读下列说明和图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

【说明】

某证券交易所为了方便提供证券交易服务，欲开发一证券交易平台，该平台的主要功能如下：

（1）开户。根据客户服务助理提交的开户信息，进行开户，并将客户信息存入客户记录中，账户信息（余额等）存入账户记录中。

（2）存款。客户可以向其账户中存款，根据存款金额修改账户余额。

（3）取款。客户可以从其账户中取款，根据取款金额修改账户余额。

（4）证券交易。客户和经纪人均可以进行证券交易（客户通过在线方式，经纪人通过电话），将交易信息存入交易记录中。

（5）检查交易。平台从交易记录中读取交易信息，将交易明细返回给客户。

现采用结构化方法对该证券交易平台进行分析与设计，获得如图 1-1 所示的上下文数据流图和图 1-2 所示的 0 层数据流图。

【问题 1】（3 分）

使用说明中的词语，给出图 1-1 中的实体 E1～E3 的名称。

【问题 2】（3 分）

使用说明中的词语，给出图 1-2 中的数据存储 D1～D3 的名称。

【问题 3】（4 分）

根据说明和图中的术语，补充图 1-2 中缺失的数据流及其起点和终点。

【问题 4】（5 分）

实际的证券交易通常是在证券交易中心完成的，因此，该平台的“证券交易”功能需将交易信息传递给证券交易中心。针对这个功能需求，需要对图 1-1 和图 1-2 进行哪些修改，请用 200 字以内的文字加以说明。

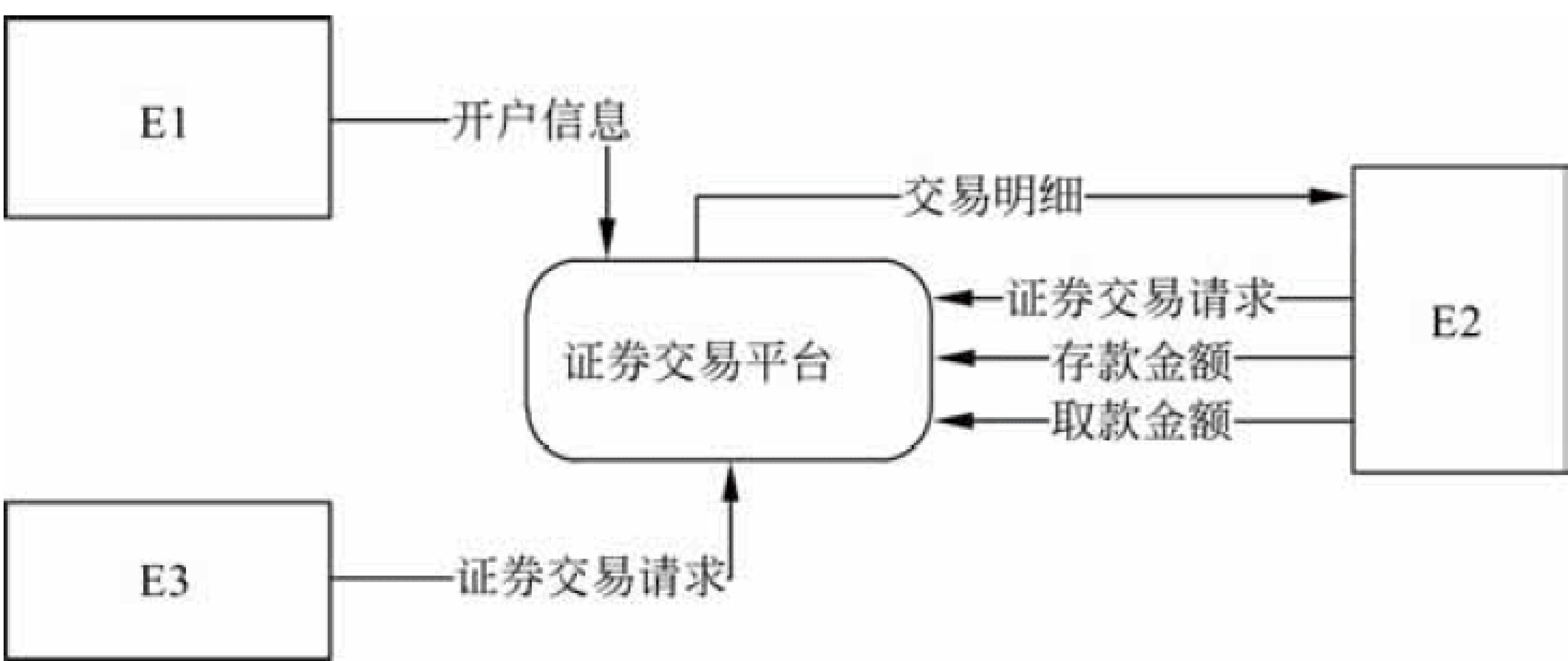


图 1-1　上下文数据流图

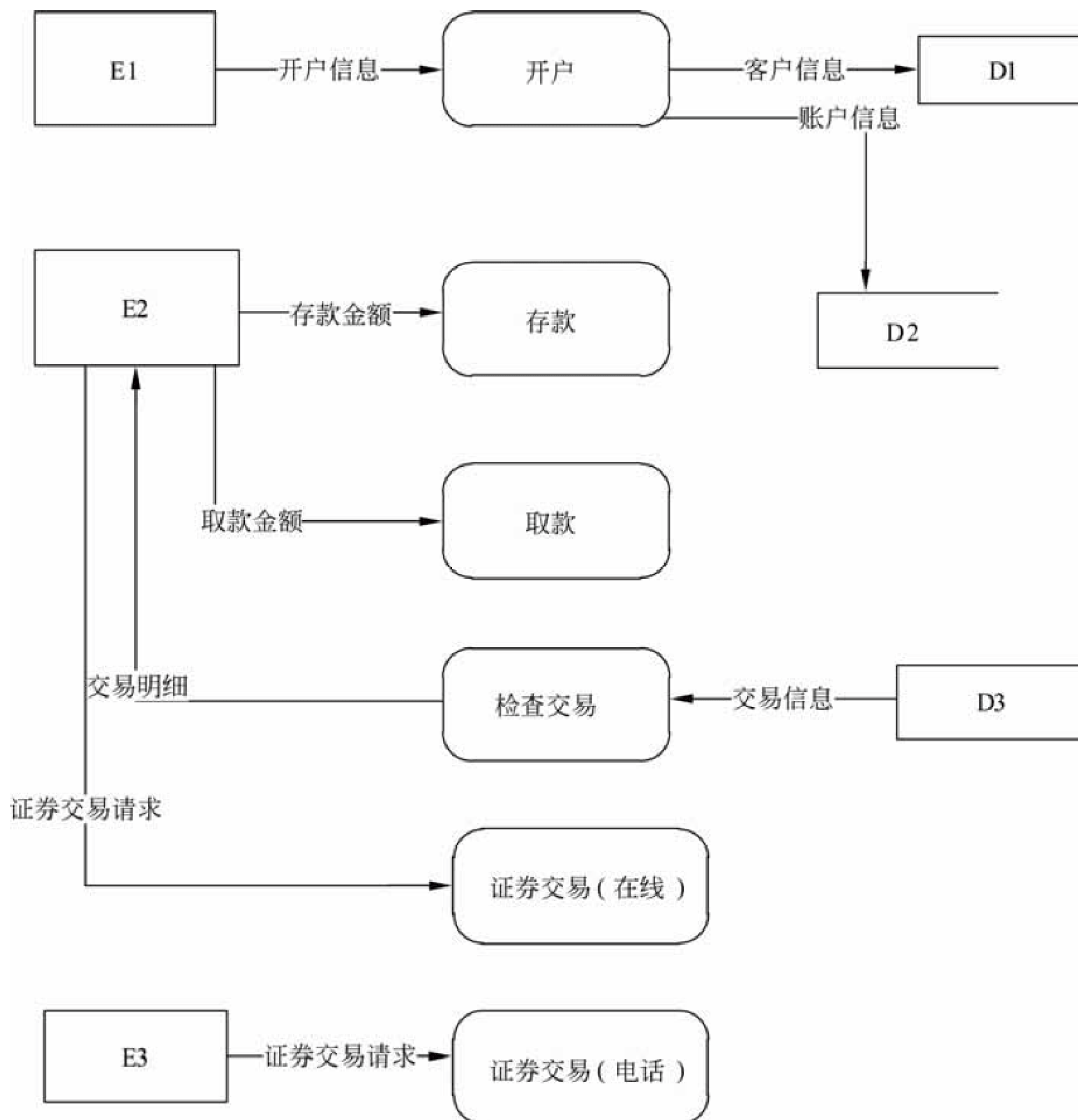


图 1-2　0 层数据流图

试题一分析

本题考查采用结构化方法进行系统分析与设计，主要考查数据流图（DFD）的应用，是传统的考题，考点与往年类似，要求考生细心分析题目中所描述的内容。本题题干描述较短，更易于分析。

DFD 是面向数据流建模的结构化分析与设计方法的重要工具，是一种便于用户理解、分析系统数据流程的图形化建模工具，是系统逻辑模型的重要组成部分。DFD 将系统建模成输入、加工（处理）、输出的模型，即流入软件的数据对象、经由加工的转换、最后以结果数据对象的形式流出软件，并采用分层的方式自顶向下建模各层数据流图，来表示不同详细程度的模型。

上下文数据流图（顶层 DFD）通常用来确定系统边界，将待开发系统看作一个大的加工，然后根据哪些外部实体为系统提供输入数据流，以及哪些外部实体接受系统发送

的数据流，建模上下文图中唯一的一个加工和一些外部实体，以及这两者之间的输入输出数据流。系统边界的变化可能使外部实体成为系统内部加工或内部加工变为外部实体。

在上下文图中确定的系统外部实体以及与外部实体的输入输出数据流的基础上，将上下文 DFD 中的加工分解成多个加工，识别这些加工的输入输出数据流，使得所有上下文 DFD 中的输入数据流，经过这些加工之后变换成上下文 DFD 的输出数据流，建模 0 层 DFD。根据 0 层 DFD 中加工的复杂程度进一步建模加工的内容。

在建模分层 DFD 时，根据需求情况可以将数据存储建模在不同层次的 DFD 中。建模时，需要注意加工和数据流的正确使用，一个加工必须既有输入又有输出；数据流必须和加工相关，即从加工流向加工、数据源流向加工或加工流向数据源。注意要在绘制下层数据流图时要保持父图与子图平衡。父图中某加工的输入输出数据流必须与它的子图的输入输出数据流在数量和名字上相同，或者父图中的一个输入（或输出）数据流对应于子图中几个输入（或输出）数据流，而子图中组成这些数据流的数据项全体正好是父图中的这一条数据流。

【问题 1】

本问题考查的是上下文 DFD，要求确定外部实体。在上下文 DFD 中，待系统名称“证券交易平台”作为唯一加工的名称，外部实体为这个唯一加工提供输入数据流或者接收其输出数据流。通过考查系统的主要功能，发现系统中涉及到客户服务助理、客户和经纪人，没有提到其他与系统交互的外部实体。根据描述（1）中“客户服务助理提交的开户信息”，（2）中“客户可以向其账户中存款”、（3）中“客户可以从其账户中取款”，（4）中“客户和经纪人均可以进行证券交易”，以及（5）中“将交易明细返回给客户”等信息，对照图 1-1，从而即可确定 E1 为“客户服务助理”实体，E2 为“客户”实体，E3 为“经纪人”实体。

【问题 2】

本问题要求确定图 1-2 中 0 层数据流图中的数据存储。重点分析说明中与数据存储有关的描述。说明（1）中“并将客户信息存入客户记录中，账户信息（余额等）存入账户记录中”，可知 D1 为客户记录、D2 为账户记录；说明（5）中“平台从交易记录中读取交易信息”，可知 D3 为交易记录。

【问题 3】

本问题要求补充缺失的数据流及其起点和终点。对照图 1-1 和图 1-2 的输入、输出数据流，数量和名称均相同，所以需要从内部确定缺失的数据流。

考查说明中的功能，先考查说明（2）/（3）中“客户可以向其账户中存款/取款，根据存款金额修改账户余额”，加工存款与取款分别需要有到数据存储账户记录（D2）标识余额的数据流，图 1-2 中加工存款与取款没有到数据存储账户记录的数据流。再考查说明（4）中“客户和经纪人均可以进行证券交易（客户通过在线方式，经纪人通过电

话），将交易信息存入交易记录中”，图 1-2 中加工证券交易（在线）和证券交易（电话）分别需要有交易记录标识交易信息的数据流。

【问题 4】

DFD 中，外部实体可以是用户，可以是其他与本系统交互的系统。如果某功能交互的是外部系统，本题中证券交易通常是在证券交易中心完成的，即证券交易中心。此时证券交易中心即为外部实体，而非本系统内部加工，因此需要对图 1-1 和图 1-2 进行修改，添加外部实体“证券交易中心”，并将数据流交易信息的终点全部改为证券交易中心。在图 1-1 中，将“证券交易中心”作为外部实体，添加从“证券交易平台”到此外部实体的数据流“交易信息”。在图 1-2 中，将“证券交易中心”作为外部实体，添加从加工“证券交易（在线）”到此外部实体的数据流“交易信息”，添加从加工“证券交易（电话）”到此外部实体的数据流“交易信息”。

试题一参考答案

【问题 1】

E1：客户服务助理　　E2：客户　　E3：经纪人

【问题 2】

D1：客户记录　　D2：账户记录　　D3：交易记录

【问题 3】

数　据　流	起　　点	终　　点
余额	存款	D2 或账户记录
余额	取款	D2 或账户记录
交易信息	证券交易（在线）	D3 或交易记录
交易信息	证券交易（电话）	D3 或交易记录

注：以上数据流与顺序无关。

【问题 4】

在图 1-1 中，将“证券交易中心”作为外部实体，添加从“证券交易平台”到此外部实体的数据流“交易信息”。

在图 1-2 中，将证券交易中心作为外部实体，添加从加工“证券交易（在线）”到此外部实体的数据流“交易信息”，添加从加工“证券交易（电话）”到此外部实体的数据流“交易信息”。

试题二（共 15 分）

阅读下列说明，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某宾馆为了有效地管理客房资源，满足不同客户需求，拟构建一套宾馆信息管理系统，以方便宾馆管理及客房预订等业务活动。

【需求分析结果】

该系统的部分功能及初步需求分析的结果如下：

（1）宾馆有多个部门，部门信息包括部门号、部门名称、电话、经理。每个部门可以有多名员工，每名员工只属于一个部门；每个部门只有一名经理，负责管理本部门。

（2）员工信息包括员工号、姓名、岗位、电话、工资，其中，员工号唯一标识员工关系中的一个元组，岗位有经理、业务员。

（3）客房信息包括客房号（如 1301、1302 等）、客房类型、收费标准、入住状态（已入住/未入住），其中客房号唯一标识客房关系中的一个元组，不同客房类型具有不同的收费标准。

（4）客户信息包括客户号、单位名称、联系人、联系电话、联系地址，其中客户号唯一标识客户关系中的一个元组。

（5）客户预订客房时，需要填写预订申请。预订申请信息包括申请号、客户号、入住时间、入住天数、客房类型、客房数量，其中，一个申请号唯一标识预订申请中的一个元组；一位客户可以有多个预订申请，但一个预订申请对应唯一的一位客户。

（6）当客户入住时，业务员根据客户的预订申请负责安排入住客房事宜。安排信息包括客房号、姓名、性别、身份证号、入住时间、天数、电话，其中客房号、身份证号和入住时间唯一标识一次安排。一名业务员可以安排多个预订申请，一个预订申请只由一名业务员安排，而且可安排多间同类型的客房。

【概念模型设计】

根据需求阶段收集的信息，设计的实体联系图如图 2-1 所示。

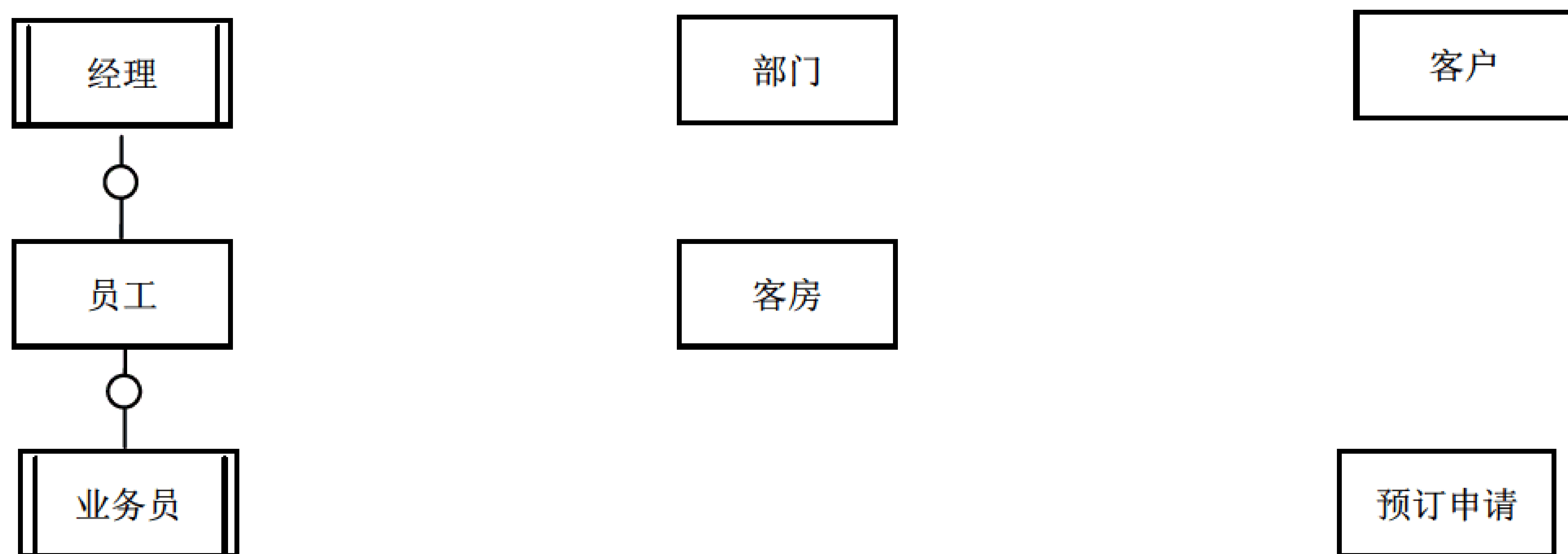


图 2-1　实体联系图

【关系模式设计】

部门（部门号，部门名称，经理，电话）

员工（员工号，____(a)____，姓名，岗位，电话，工资）

客户（____(b)____，联系人，联系电话，联系地址）

客房（客房号，客房类型，收费标准，入住状态）

预订申请（____(c)____，入住时间，天数，客房类型，客房数量）

安排（申请号，客房号，姓名，性别，____(d)____，天数，电话，业务员）

【问题 1】（4 分）

根据问题描述，补充四个联系，完善图 2-1 的实体联系图。联系名可用联系 1、联系 2、联系 3 和联系 4 代替，联系的类型为 1:1、1:n 和 m:n（或 1:1、1:*和*:*）。

【问题 2】（8 分）

（1）根据题意，将关系模式中的空（a）～（d）补充完整，并填入答题纸对应的位置上。

（2）给出“预订申请”和“安排”关系模式的主键和外键。

【问题 3】（3 分）

【关系模式设计】中的“客房”关系模式是否存在规范性问题，请用 100 字以内文字解释你的观点（若存在问题，应说明如何修改“客房”关系模式）。

试题二分析

本题考查数据库系统中实体联系模型(E-R 模型)和关系模式设计方面的基础知识。

【问题 1】

① 根据题意“每个部门可以有多名员工，每名员工只属于一个部门”，所以部门和员工之间有一个“所属”联系，联系类型为 1:*。

② 根据题意“每个部门有一名经理，只负责管理本部门的事务”，所以部门和经理之间有一个“负责”联系，联系类型为 1:1。

③ 根据题意“一个客户可以有多个预订申请，但一个预订申请对应唯一的一个客户号”，所以客户和预订申请之间有一个“预订”联系，联系类型为 1:*。

④ 根据题意“一个业务员可以安排多个预订申请，一个预订申请只由一个业务员安排，而且可安排多个同类型的客房”，即一份预订申请可以预订多间同类型的客房，所以业务员与客房和预订申请之间的“安排”联系类型为 1:*:*。

根据上述分析，完善图 2-1 所示的实体联系图如图 2-2 所示。

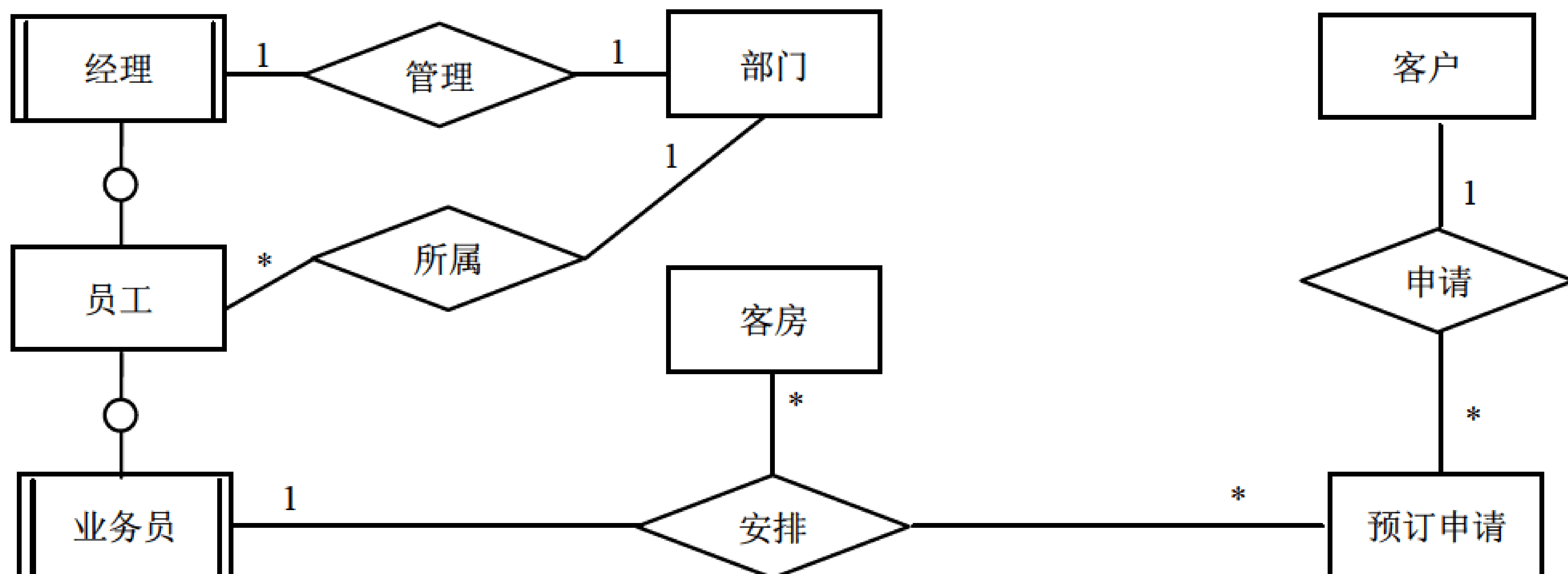


图 2-2 完善的实体联系图

【问题 2】

由于部门和员工之间有一个$1-2^{-8}$的“所属”联系需要将一端的码“部门号”并入多端，故员工关系模式中的空（a）应填写“部门号”。

根据题意，客户信息包括客户号、单位名称、联系人、联系电话、联系地址，给定的客户关系模式中，不含客户号、单位名称，故空（b）应填写“客户号，单位名称”。

由于预订申请信息包括申请号、客户号、预订入住时间、入住天数、客房类型、客房数量，故空（c）应填写“申请号，客户号”。

根据题意“客房号、身份证号和入住时间唯一标识安排联系的每一个元组”，所以空（d）应填写“身份证号，入住时间”。

根据题意，“一个申请号对应唯一标识预订申请中的每一个元组”，所以预订申请关系模式的主键为申请号；又因为客户号是客户关系的主键，根据外键定义可知，客户号是预订申请关系的外建。

根据题意“客房号、身份证号和入住时间唯一标识安排联系的每一个元组”，所以安排关系模式的主键为客房号，身份证号，入住时间；外键为申请号，客房号，业务员，因为申请号和客房号为预约申请和客房关系的主键，而“业务员”是员工关系子实体必须参考员工关系的主键“员工号”，所以业务员也是外键。

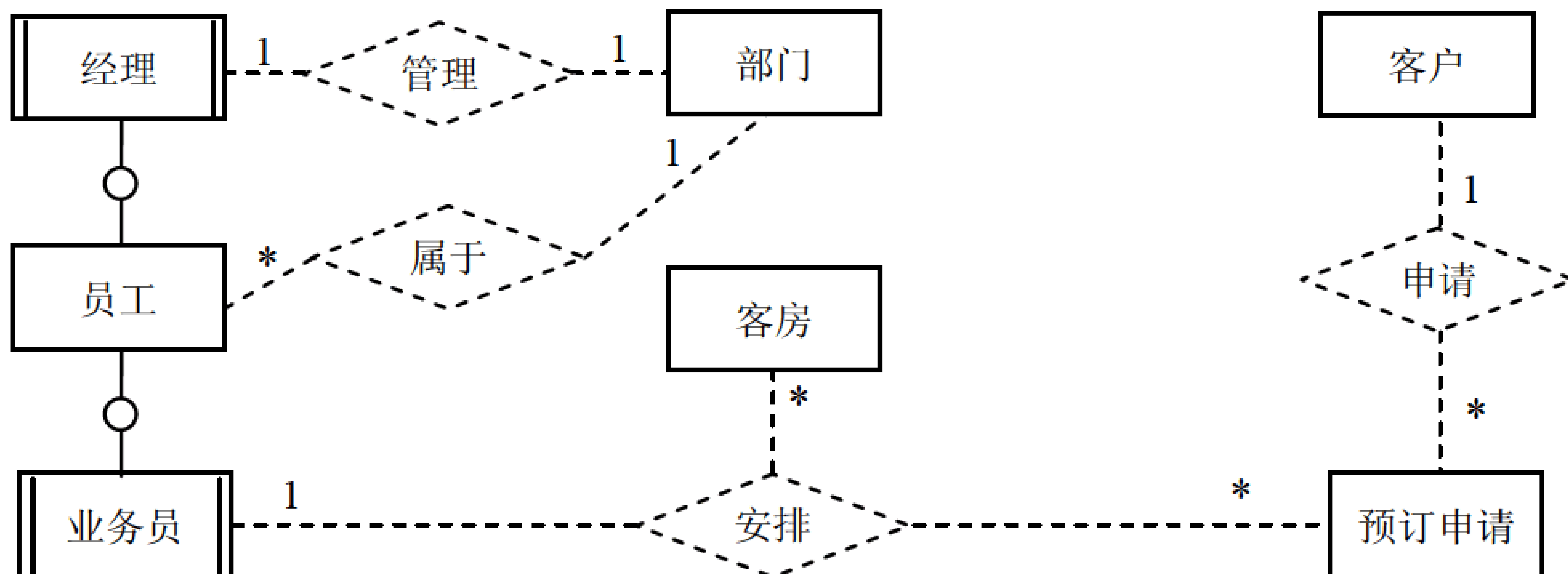
【问题 3】

客房关系模式存在问题。因为客房号为主键，所以客房号可以决定全属性，即客房号→（客房类型，收费标准，入住状态）。又因为客房类型→收费标准，所以该关系模式存在传递依赖，没有达到 3NF，应将客房关系模式分解为客房 1（客房号，客房类型，入住状态），客房 2（客房类型，收费标准）。

试题二参考答案

【问题 1】

完善后的实体联系图如下所示（所补充的联系和类型如虚线所示）：



（注：*可以用 m、n 表示）

【问题 2】

（1）

（a）部门号

（b）客户号，单位名称

（c）申请号，客户号

（d）身份证号，入住时间

（2）

“预订申请”关系模式：主键为申请号
　　　　　　　　　　外键为客户号

“安排”关系模式：主键为客房号，身份证号，入住时间
　　　　　　　　外键为申请号，客房号，业务员

【问题 3】

存在问题。

关系模式存在传递依赖，没有达到 3NF。

应将客房关系模式分解为客房 1（客房号，客房类型，入住状态）客房 2（客房类型，收费标准）。

试题三（共 15 分）

阅读下列说明，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

【说明】

某种出售罐装饮料的自动售货机（Vending Machine）的工作过程描述如下：

（1）顾客选择所需购买的饮料及数量。

（2）顾客从投币口向自动售货机中投入硬币（该自动售货机只接收硬币）。硬币器收集投入的硬币并计算其对应的价值。如果所投入的硬币足够购买所需数量的这种饮料且饮料数量足够，则推出饮料，计算找零，顾客取走饮料和找回的硬币；如果投入的硬币不够或者所选购的饮料数量不足，则提示用户继续投入硬币或重新选择饮料及数量。

（3）一次购买结束之后，将硬币器中的硬币移走（清空硬币器），等待下一次交易。

自动售货机还设有一个退币按钮，用于退还顾客所投入的硬币。已经成功购买饮料的钱是不会被退回的。

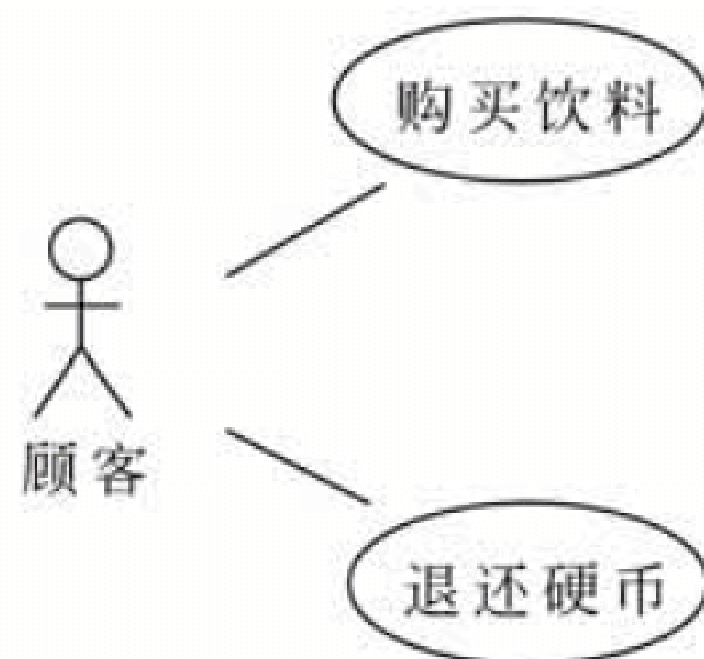


图 3-1　用例图

现采用面向对象方法分析和设计该自动售货机的软件系统，得到如图 3-1 所示的用例图，其中，用例“购买饮料”的用例规约描述如下。

参与者：顾客。

主要事件流：

1. 顾客选择需要购买的饮料和数量，投入硬币；
2. 自动售货机检查顾客是否投入足够的硬币；
3. 自动售货机检查饮料储存仓中所选购的饮料是否足够；
4. 自动售货机推出饮料；
5. 自动售货机返回找零。

备选事件流：

2a．若投入的硬币不足，则给出提示并退回到 1；

3a．若所选购的饮料数量不足，则给出提示并退回到 1。

根据用例“购买饮料”得到自动售货机的 4 个状态：“空闲”状态、“准备服务”状态、“可购买”状态以及“饮料出售”状态，对应的状态图如图 3-2 所示。

所设计的类图如图 3-3 所示。

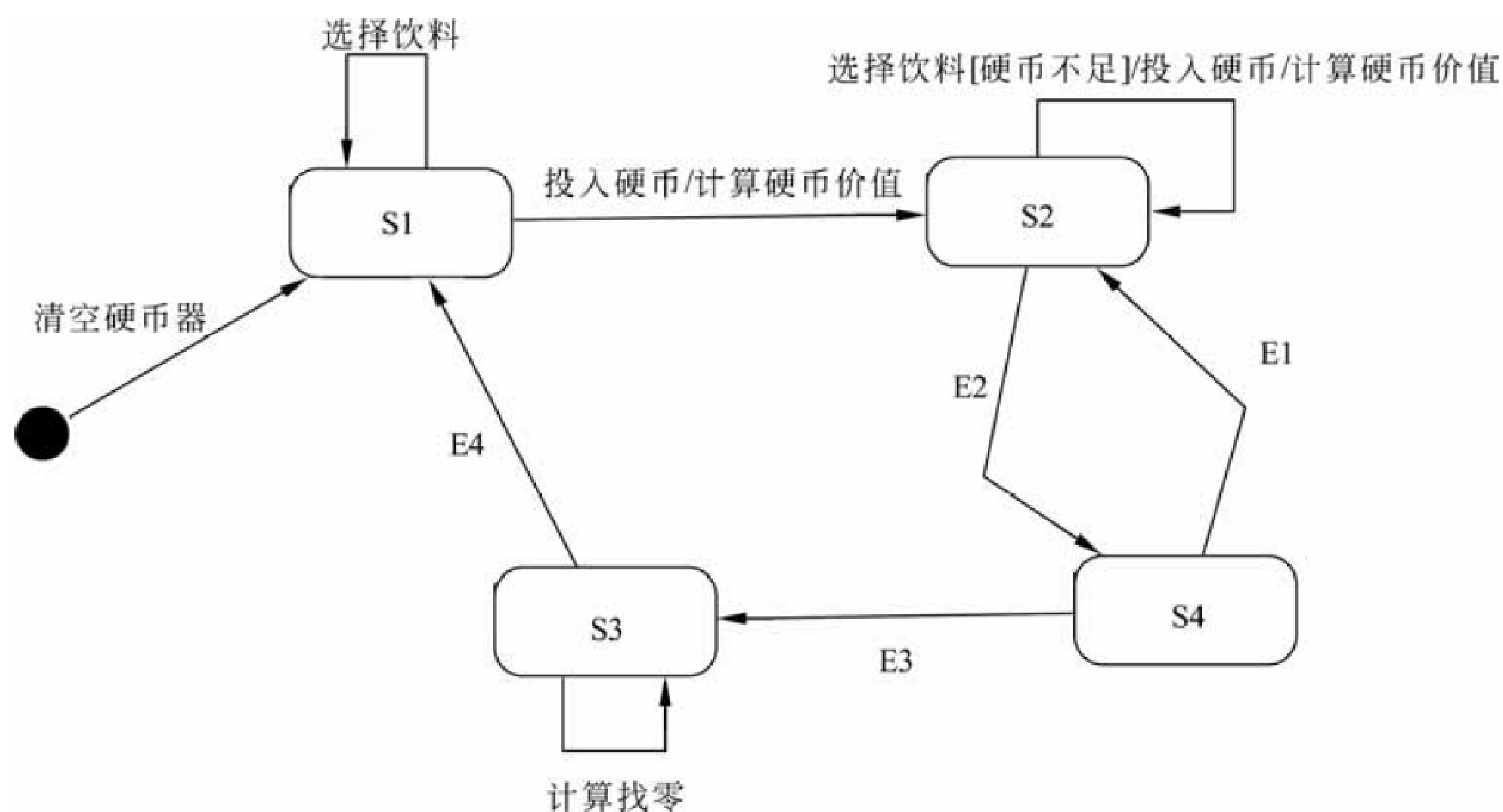


图 3-2　状态图

【问题 1】（6 分）

根据说明中的描述，使用说明中的术语，给出图 3-2 中的 S1～S4 所对应的状态名。

【问题 2】（4 分）

根据说明中的描述，使用说明中的术语，给出图 3-2 中 E1～E4 所对应的事件名称。

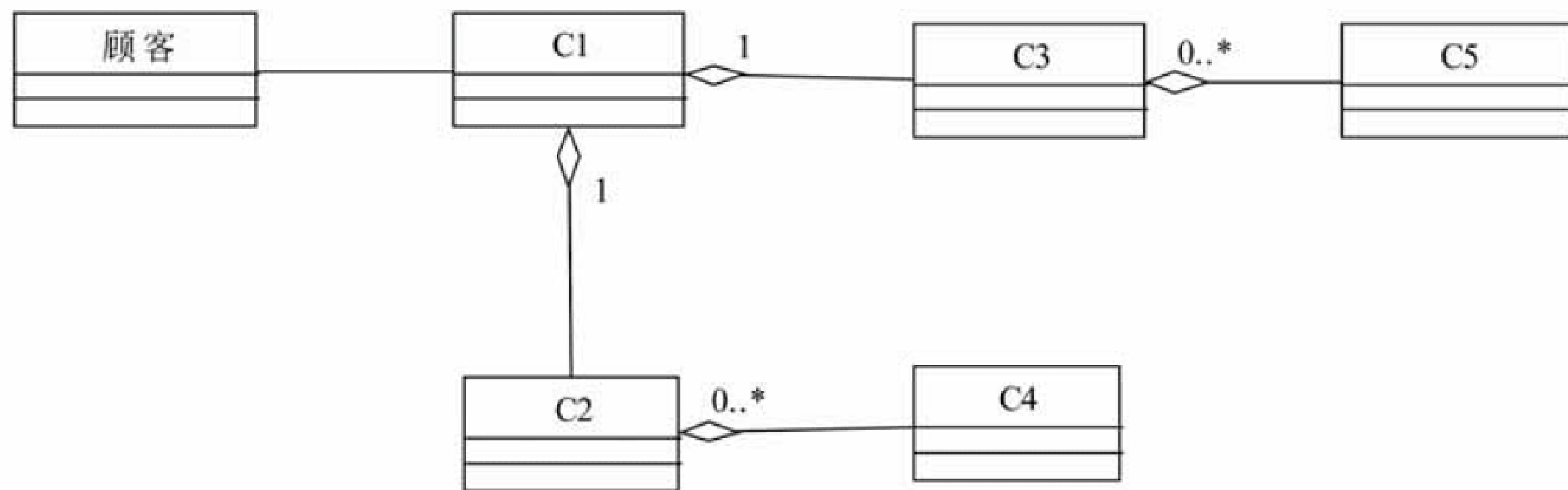


图 3-3 类图

【问题 3】（5 分）

根据说明中的描述，使用说明中的术语，给出图 3-3 中 C1～C5 所对应的类名。

试题三分析

本题属于经典的考题，主要考查面向对象分析与设计的基本概念。在建模方面，本题涉及到了用例图、状态图和类图。用例是描述系统功能需求的一种常用方法，用例规约是创建需求模型，进行系统设计的依据。本题的考点就是由用例规约创建状态图和类图。

【问题 1】

题目说明中已经给出了自动售货机的 4 个状态分别是：“空闲”状态、“准备服务”状态、“可购买”状态以及“饮料出售”状态。解答本题需要根据用例规约推出这 4 个状态之间的迁移关系，这样才能与图 3-2 中的状态 S1～S4 对应。

首先从状态图的初始状态“●”开始，S1 代表的就是自动售货机的初始状态。在上述 4 个状态中，只有在“空闲”下，才能开始一次售卖，所以 S1 对应的是“空闲”状态。

根据 S2 相关的事件来看，在该状态时，自动售货机在接收顾客的请求（顾客选择的饮料以及投入的硬币），因此应对应“准备服务”状态。

状态 S3 有一个自迁移事件“计算找零”，根据说明和用例规约可知，饮料出售之后进行找零，所以 S3 对应“饮料出售”状态。S4 则对应“可购买”状态。

【问题 2】

确定了状态图中的各个状态，接下来就需要进行状态之间迁移事件的获取。E2 是从“准备服务”状态变换到“可购买”状态的事件，“选择饮料[硬币不足]”时仍然停留在“准备服务”状态，对应用例规约中的 2a；根据用例规约若硬币足够则进入下一步，所以 E2 所对应的事件是“选择饮料[硬币足够购买饮料]”。

E1 事件的触发将使得自动售货机从“可购买”状态变换到“准备服务”状态，对应用例规约中的 3a，所以 E1 对应的事件应是“所选购的饮料数量不足”。

E3 事件的触发将使得自动售货机从“可购买”状态迁移到“饮料出售”状态。根据说明，能够售出饮料必须满足两个条件：该饮料数量足够以及顾客投入的硬币足够。硬币是否足够以及饮料数量不足在状态 S2 已经进行了判断，因此 E3 对应的事件应是“所选购的饮料数量足够/推出饮料”。

E4 对应的事件是自动售货机完成一次售卖，回到“空闲”状态时需处理的事件，根据说明可知，E4 对应的事件应是“取走饮料/找零并清空硬币器”。

【问题 3】

本题要求根据说明和用例规约创建对应的类模型。根据说明和用例规约可知自动售货机有几个重要的组成元素：饮料、硬币、硬币器和饮料存储仓。1 台自动售货机有 1 个硬币器、1 个饮料存储仓；硬币器可以接收多枚硬币，饮料存储仓中可以容纳多种饮料。由此可知，图 3-3 中的两个 0..*聚集关系应该分别对应“硬币器-硬币”和“饮料存储仓-饮料”这两对“部分-整体”关系；而 C1 就是自动售货机。

试题三参考答案

【问题 1】

S1：空闲　　　　S2：准备服务

S3：饮料出售　　S4：可购买

【问题 2】

E1：所选购的饮料数量不足

E2：选择饮料[硬币足够购买饮料]

E3：所选购的饮料数量足够/推出饮料

E4：取走饮料/找零并清空硬币器

【问题 3】

C1：自动售货机　　C2：硬币器　　C3：饮料存储仓

C4：硬币　　C5：饮料

或者

C1：自动售货机　　C2：饮料存储仓　　C3：硬币器

C4：饮料　　C5：硬币

试题四（共 15 分）

阅读下列说明和 C 代码，回答问题 1 至问题 3，将解答写在答题纸的对应栏内。

【说明】

模式匹配是指给定主串 t 和子串 s，在主串 t 中寻找子串 s 的过程，其中 s 称为模式。如果匹配成功，返回 s 在 t 中的位置，否则返回−1。

KMP 算法用 next 数组对匹配过程进行了优化。KMP 算法的伪代码描述如下：

1. 在串 t 和串 s 中，分别设比较的起始下标 i=j=0。
2. 如果串 t 和串 s 都还有字符，则循环执行下列操作：

（1）如果 j=-1 或者 t[i]=s[j]，则将 i 和 j 分别加 1，继续比较 t 和 s 的下一个字符；

（2）否则，将 j 向右滑动到 next[j]的位置，即 j=next[j]。

3. 如果 s 中所有字符均已比较完毕，则返回匹配的起始位置（从 1 开始）；否则返回−1。其中，next 数组根据子串 s 求解。求解 next 数组的代码已由 get_next 函数给出。

【C 代码】

（1）常量和变量说明

t，s：长度为 lt 和 ls 的字符串
next：next 数组，长度为 ls

（2）C 程序

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
 /*求 next[]的值*/
void get_next( int *next, char *s, int ls )  {
     int i = 0, j = -1 ;
     next[0] = -1 ; /*初始化 next[0] */
     while ( i < ls ){ /*还有字符*/
        if( j == -1 || s[i] == s[j] ){/*匹配*/
             j++ ;
             i++ ;
             if( s[i] == s[j] )
                next[i] = next[j];
             else
                next[i] = j ;

          }
          else
             j = next[j] ;
     }
 }
int kmp( int *next, char *t ,char *s, int lt, int ls )
 {
     int i = 0, j = 0 ;
     while ( i < lt &&____(1)____){
        if( j == -1 || ____(2)____){
             i ++ ;
             j ++ ;
```

```
            } else
                  ___(3)___ ;
        }
        if ( j >= ls)
            return ___(4)___ ;
        else
            return -1 ;
    }
```

【问题 1】（8 分）

根据题干说明，填充 C 代码中的空（1）～（4）。

【问题 2】（2 分）

根据题干说明和 C 代码，分析出 KMP 算法的时间复杂度为__（5）__（主串和子串的长度分别为 lt 和 ls，用 O 符号表示）。

【问题 3】（5 分）

根据 C 代码，字符串“BBABBCAC”的 next 数组元素值为__（6）__（直接写元素值，之间用逗号隔开）。若主串为“AABBCBBABBCACCD”，子串为“BBABBCAC”，则函数 kmp 的返回值是__（7）__。

试题四分析

本题考查算法设计与分析以及用 C 程序设计语言实现算法的能力。

KMP 算法是一个非常经典的模式匹配算法。其核心思想是核心思想：匹配过程中字符对不相等时，不需回溯主串，而是利用已经得到的部分匹配结果将模式向右滑动尽可能远的一段距离继续比较。滑动的距离由 next 数组给出。该算法提出之后，有一些改进的思想，使得 next 数组的计算有多种方式。本题干不需要考生考虑如何计算 next 数组，已经直接给出计算该数组的 C 代码。只需要根据已经计算的 next 数组进行模式匹配即可。

【问题 1】

在 C 函数 kmp 中，while 循环是判断串 s 和 t 是否还有字符，因此空（1）处应填写“j < ls”。根据题干描述，“如果 j=-1 或者 t[i]=s[j]，则将 i 和 j 分别加 1”，则空（2）处填入“t[i] == s[j]”，空（3）处是“否则，将 j 向右滑动到 next[j]的位置，即 j=next[j]”的情况，因此填入“j = next[j]”。空（4）处要填返回值，此处应该是能找到模式串的情况，此时 i 是主串匹配完成后的位置，j 是子串的长度，则匹配的起始位置为 i–j+1（从 1 开始）。

【问题 2】

在 kmp 函数中，只有一个 while 循环，该算法的时间复杂度为 O(lt+ls)。

【问题 3】

根据 C 函数 get_next，得到“BBABBCAC”的 next 数组的值为–1，–1，1，–1，–1，

2，0，0。对主串为“AABBCBBABBCACCD”和上述模式串，得到匹配位置为 6，这里需要注意的是，位置从 1 开始。

试题四参考答案

【问题 1】

（1）j＜ls　　（2）t[i] == s[j]　　（3）j = next[j]　　（4）i-j+1

【问题 2】

（5）O(lt+ls)

【问题 3】

（6）−1，−1，1，−1，−1，2，0，0

（7）6

试题五（共 15 分）

阅读下列说明和 C++代码，将应填入(n)处的字句写在答题纸的对应栏内。

【说明】

某发票（Invoice）由抬头（Head）部分、正文部分和脚注（Foot）部分构成。现采用装饰（Decorator）模式实现打印发票的功能，得到如图 5-1 所示的类图。

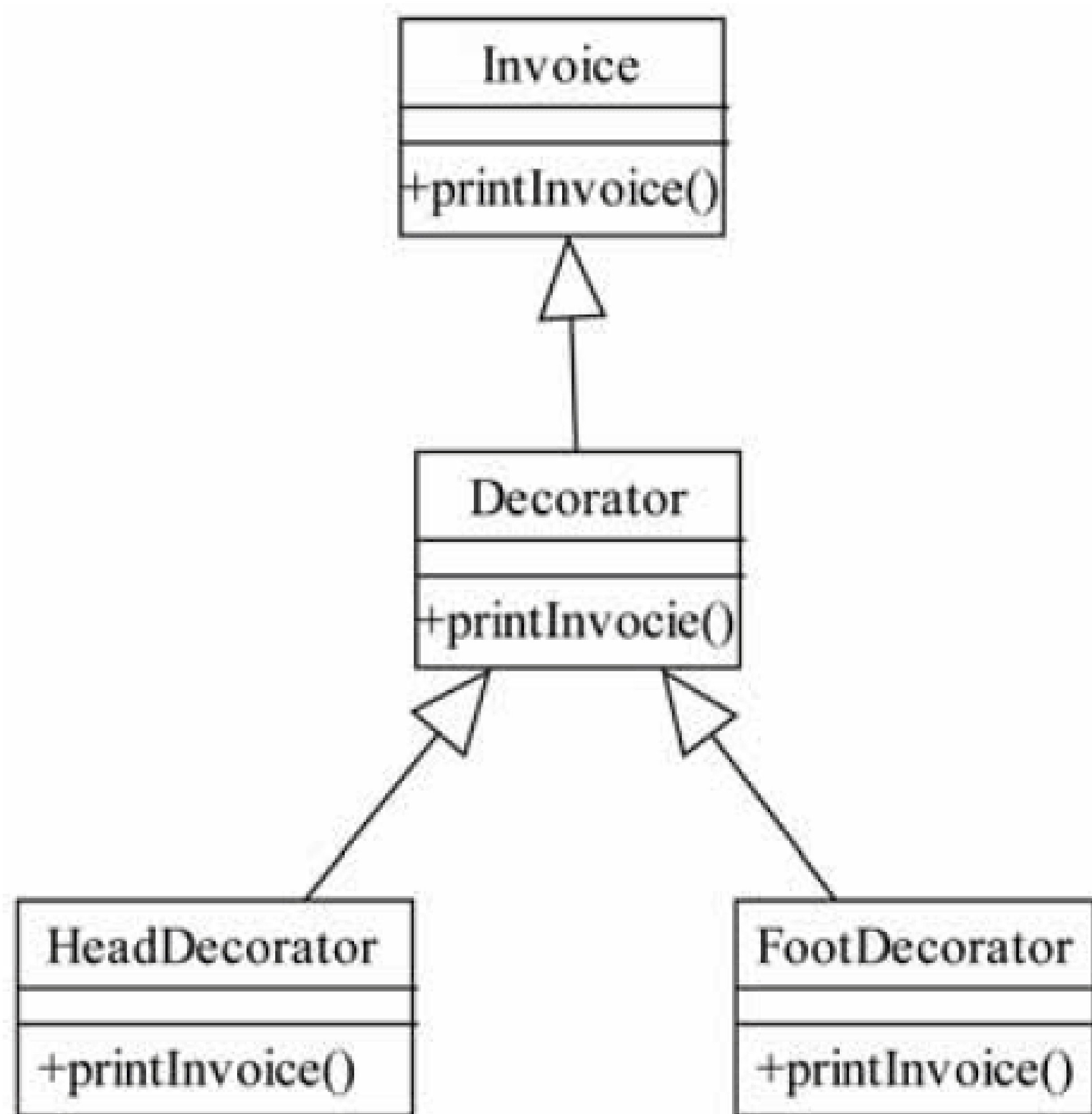


图 5-1　类图

【C++代码】

```
#include <iostream>
using namespace std;
class Invoice {
public:
    ____(1)____ {
        cout << "This is the content of the invoice!" << endl;
```

```
    }
};
class Decorator : public Invoice{
    Invoice *ticket;
public:
    Decorator(Invoice *t)  { ticket = t; }
    void printInvoice(){
        if(ticket != NULL)
            ____(2)____;
    }
};
class HeadDecorator : public Decorator{
public:
    HeadDecorator(Invoice *t): Decorator(t){ }
    void printInvoice() {
        cout << "This is the header of the invoice!" << endl;
        ____(3)____;
    }
};
class FootDecorator : public Decorator{
 public:
    FootDecorator(Invoice *t): Decorator(t) { }
    void printInvoice() {
        ____(4)____;
        cout << "This is the footnote of the invoice!" << endl;
    }
};
int main(void) {
    Invoice t;
    FootDecorator f(&t);
    HeadDecorator h(&f);
    h.printInvoice();
    cout << "-------------------------"<< endl;
    FootDecorator a(NULL);
    HeadDecorator b( ____(5)____ );
    b.printInvoice();
    return 0;
}
```

程序的输出结果为：

```
This is the header of the invoice!
```

```
This is the content of the invoice!
This is the footnote of the invoice!
------------------------
This is the header of the invoice!
This is the footnote of the invoice!
```

试题五分析

本题考查装饰（Decorator）模式的基本概念和应用。

装饰模式属于结构型设计模式，其设计意图是动态地给一个对象添加一些额外的职责。就增加功能而言，装饰模式比生成子类更加灵活。装饰模式的结构如图 5-2 所示。

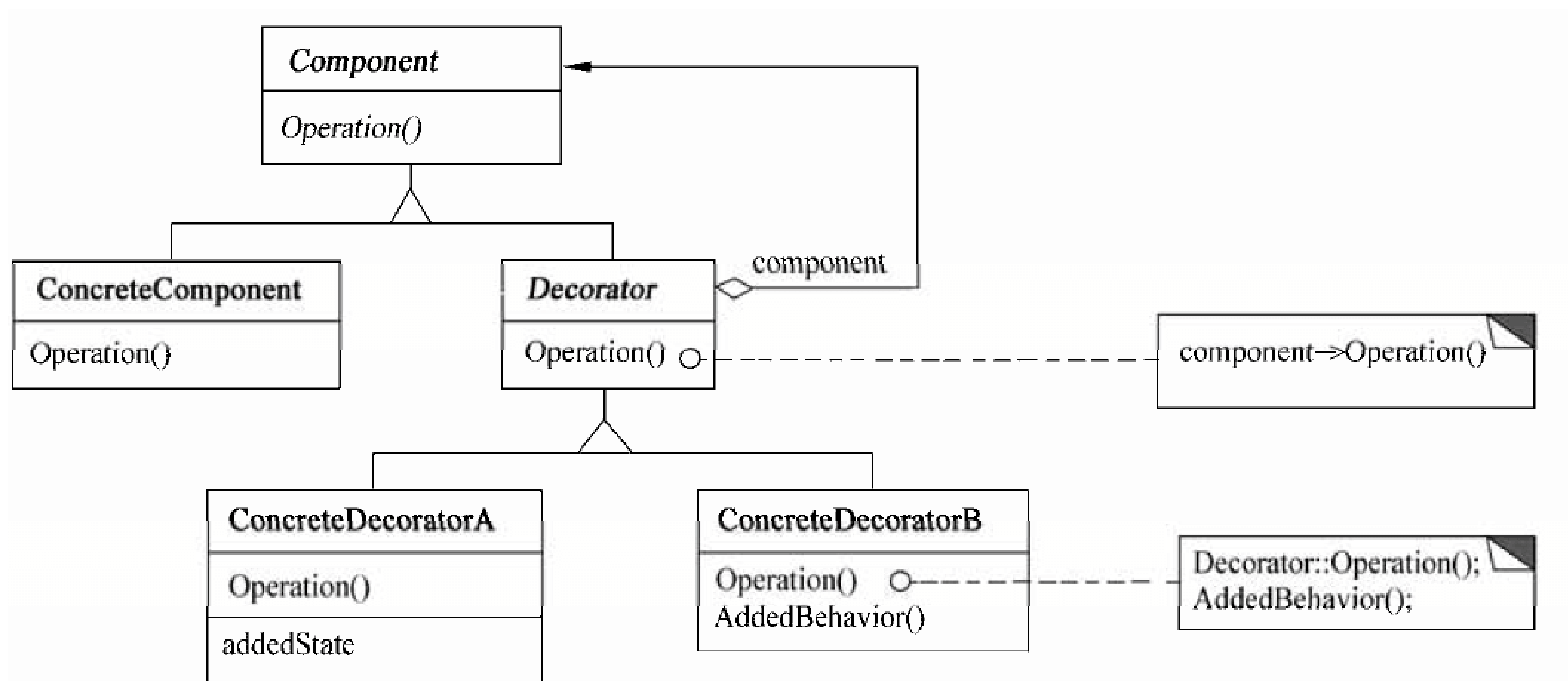


图 5-2　装饰模式结构图

其中：

- Component 定义一个对象接口，可以给这些对象动态地添加职责。
- ConcreteComponent 定义一个对象，可以给这个对象添加一些职责。
- Decorator 维持一个指向 Component 对象的指针，并定义一个与 Component 接口一致的接口。
- ConcreteDecorator 向组件添加职责。

装饰模式适用于：

- 在不影响其他对象的情况下，以动态、透明的方式给单个对象添加职责。
- 处理那些可以撤销的职责。
- 当不能采用生成子类的方式进行扩充时。一种情况是，可能有大量独立的扩展，为支持每一种组合将产生大量的子类，使得子类数目呈爆炸性增长。另一种情况可能是，由于类定义被隐藏，或类定义不能用于生成子类。

本题将装饰模式用于实现打印发票问题。图 5-1 的类图中，类 Invoice 对应图 5-2 中

的 Component，其功能是打印发票的内容；HeadDecorator 和 FootDecorator 是两个 ConcreteDecorator，向组件中添加打印发票抬头和发票脚注的功能。

方法 printInvoice 是 Invoice 中定义的接口，Component 类中应定义一个与之一致的接口。在 C++中，父类和子类之间共享接口，通常采用虚拟函数。由此可知，空（1）处应填写“virtual void printInvoice()”。这个接口在类 Decorator、HeadDecorator 和 FootDecorator 中分别进行了重置，分别对应代码中的空（2）～（4）。

类 Decorator 中保持了一个指向 Component 对象的指针——ticket，用来接收所要装饰的组件 Invoice。因此空（2）处应填写“ticket->printInvoice()”。类 HeadDecorator 和 FootDecorator 是在打印发票内容的基础上，打印发票的抬头和脚注，所以空（3）、（4）处都应填写“Decorator::printInvoice()”。

最后一空考查的是装饰模式的调用，由 main()函数中给出的第一次调用可以获得一些提示，推断出空（5）出应填写“&a”。

试题五参考答案

（1）virtual void printInvoice()

（2）ticket->printInvoice()

（3）Decorator::printInvoice()

（4）Decorator::printInvoice()

（5）&a

试题六（共 15 分）

阅读下列说明和 Java 代码，将应填入<u>（n）</u>处的字句写在答题纸的对应栏内。

【说明】

某发票（Invoice）由抬头（Head）部分、正文部分和脚注（Foot）部分构成。现采用装饰（Decorator）模式实现打印发票的功能，得到如图 6-1 所示的类图。

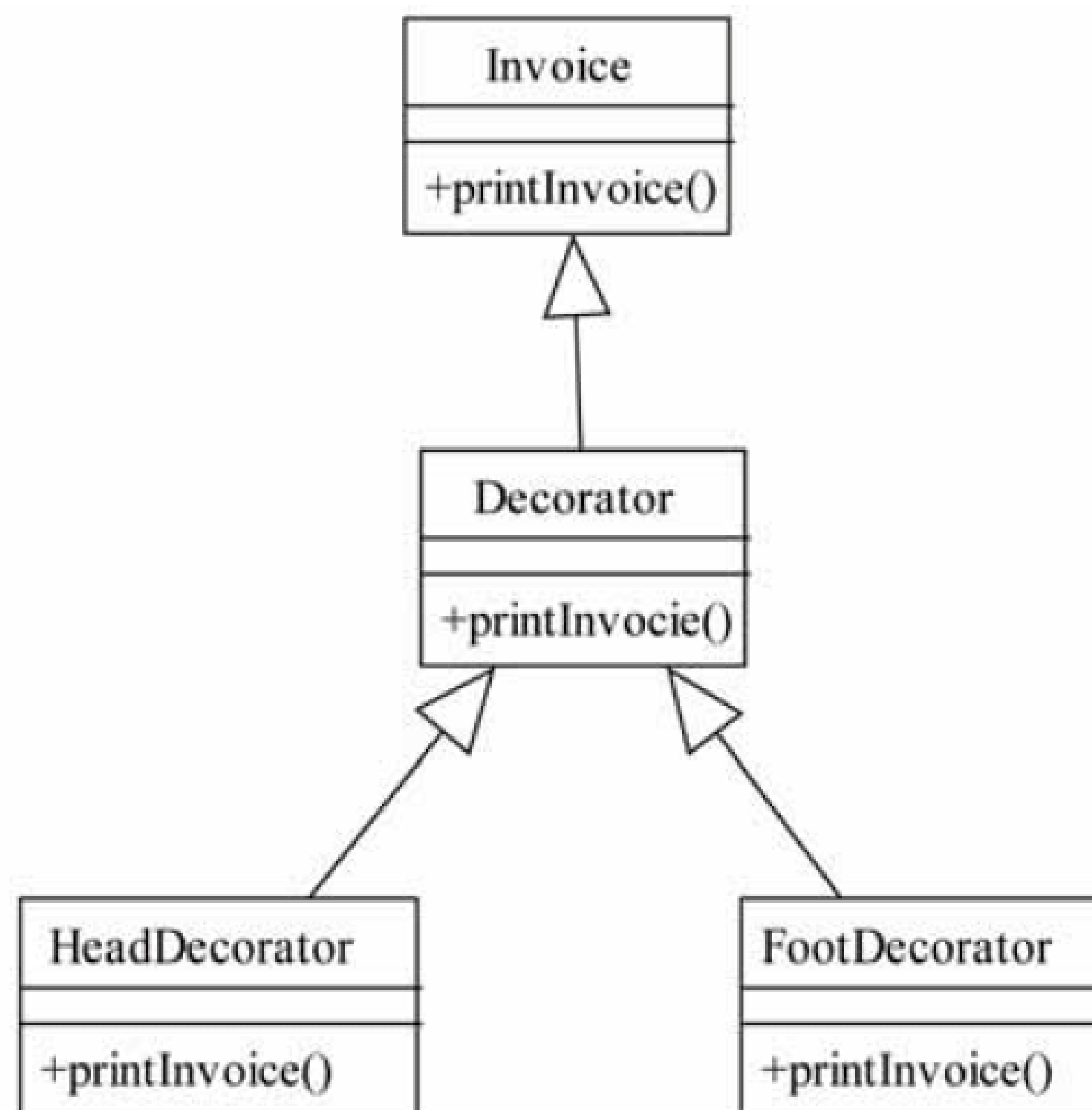


图 6-1　类图

【Java 代码】

```
class Invoice {
    public void printInvoice() {
        System.out.println("This is the content of the invoice!");
    }
}

class Decorator extends Invoice{
    protected Invoice ticket;
    public Decorator(Invoice t){
        ticket = t;
    }
    public void printInvoice(){
        if(ticket != null)
            ____(1)____;
    }
}
class HeadDecorator extends Decorator{
    public HeadDecorator(Invoice t) {
        super(t);
    }
    public void printInvoice() {
        System.out.println("This is the header of the invoice!");
        ____(2)____;
    }
}
class FootDecorator extends Decorator{
    public FootDecorator(Invoice t) {
        super(t);
    }
    public void printInvoice() {
        ____(3)____;
        System.out.println("This is the footnote of the invoice!");
    }
}
class Test {
    public static void main(String[] args) {
        Invoice t = new Invoice();
        Invoice ticket;
        ticket = ____(4)____;
```

```
            ticket.printInvoice();
            System.out.println("------------------------");
            ticket = ____(5)____;
            ticket.printInvoice();
        }
    }
```

程序的输出结果为：

```
This is the header of the invoice!
This is the content of the invoice!
This is the footnote of the invoice!
------------------------
This is the header of the invoice!
This is the footnote of the invoice!
```

试题六分析

本题考查装饰（Decorator）模式的基本概念和应用。

装饰模式属于结构型设计模式，其设计意图是动态地给一个对象添加一些额外的职责。就增加功能而言，装饰模式比生成子类更加灵活。装饰模式的结构如图 6-2 所示。

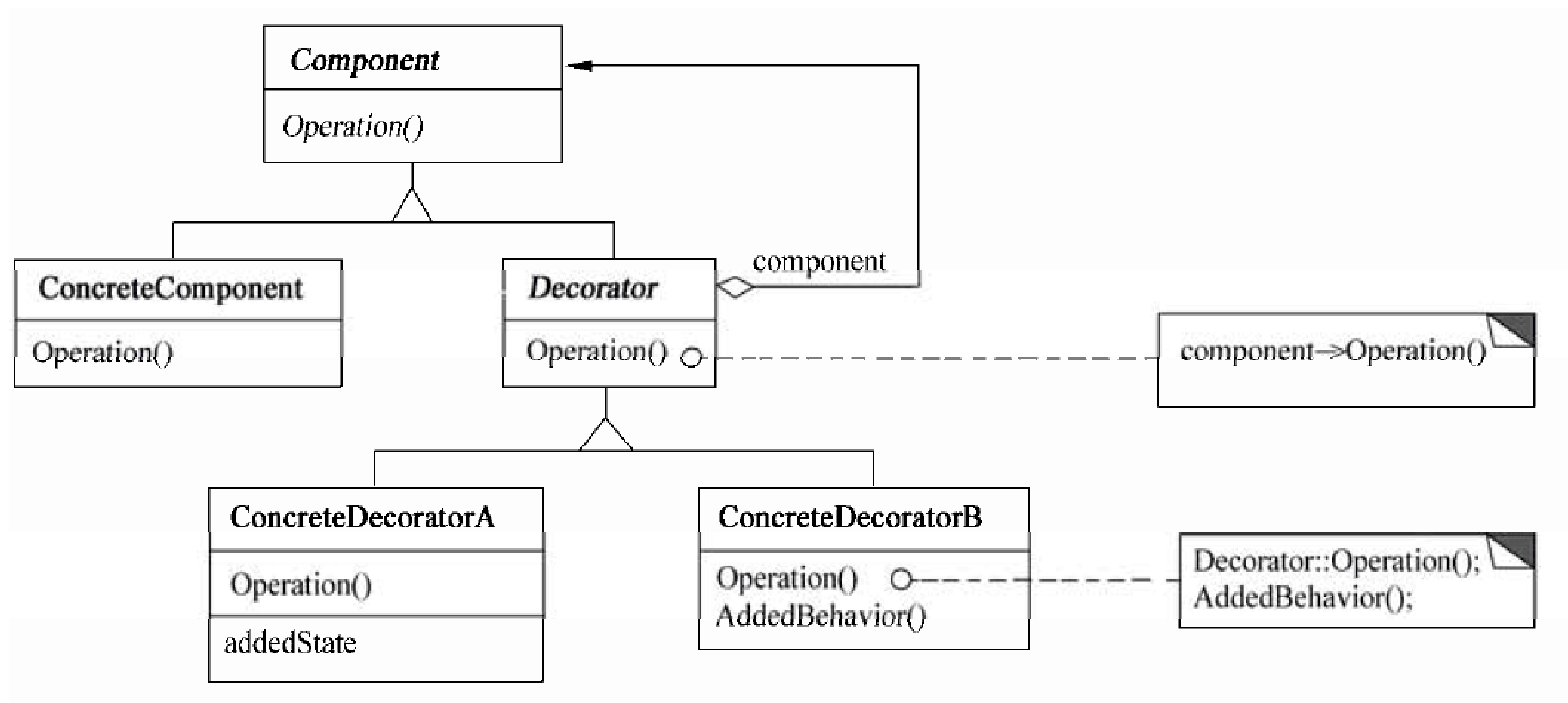


图 6-2　装饰模式结构图

其中：

- Component 定义一个对象接口，可以给这些对象动态地添加职责。
- ConcreteComponent 定义一个对象，可以给这个对象添加一些职责。
- Decorator 维持一个指向 Component 对象的指针，并定义一个与 Component 接口一致的接口。

- ConcreteDecorator 向组件添加职责。

装饰模式适用于：

- 在不影响其他对象的情况下，以动态、透明的方式给单个对象添加职责。
- 处理那些可以撤销的职责。
- 当不能采用生成子类的方式进行扩充时。一种情况是，可能有大量独立的扩展，为支持每一种组合将产生大量的子类，使得子类数目呈爆炸性增长。另一种情况可能是，由于类定义被隐藏，或类定义不能用于生成子类。

本题将装饰模式用于实现打印发票问题。图 6-1 的类图中，类 Invoice 对应图 6-2 中的 Component，其功能是打印发票的内容；HeadDecorator 和 FootDecorator 是两个 ConcreteDecorator，向组件中添加打印发票头和发票脚注的功能。

方法 printInvoice 是 Invoice 中定义的接口，Component 类中应定义一个与之一致的接口。这个接口在类 Decorator、HeadDecorator 和 FootDecorator 中分别进行了重新定义，分别对应代码中的空（1）～（3）。

类 Decorator 中保持了一个 Component 对象——ticket，用来接收所要装饰的组件 Invoice。因此空（1）处应填写“ticket.printInvoice()”。类 HeadDecorator 和 FootDecorator 是在打印发票内容的基础上，打印发票的抬头和脚注，所以空（2）、（3）处都应填写“super.printInvoice()”。

空（4）～（5）考查的是装饰模式的调用，分别应填写为“new HeadDecorator(new FootDecorator(t))”和“new HeadDecorator(new FootDecorator(null))”。

试题六参考答案

（1）ticket.printInvoice()

（2）super.printInvoice()

（3）super.printInvoice()

（4）new HeadDecorator(new FootDecorator(t))

（5）new HeadDecorator(new FootDecorator(null))